LINUX의 세계에로

Red Hat Linux 7.1 배우기

김책공업종합대학출판사
주체91

Red Hat Linux 7.1

배 우 기

김책공업종합대학출판사

차 례

제 5 편. 기본접속

제 6 편. 소규모사무/가정사무에서 Linux 의 리용

제 7 편. Red Hat Linux 7.1 을 Web 와 전자우편봉사기로 리용

부록

머 리 말

탁상형개인용콤퓨터에서 Microsoft 회사의 조작체계들을 대신할수 있는것들중의 하나로서 Linux 가 등장했는데 이것은 여기에 취미를 가진 사람들이 재정적인 영향을 받지 않고도 실행해 볼수 있는것으로 해서 환영을 받고 있다.

Linux 는 가정용콤퓨터에 Unix 의 성능(이전에는 값 비싼 봉사기들과 워크스테이션에만 해당)을 제공하는 빠르고 효률적인 조작체계이며 자유롭게 배포할수 있다. 이것은 넓은 범위의 과제(기관정보체계하부구조의 중요한 사명을 수행하는 Web 싸이트로부터 교육용응용프로그람에 이르기까지)를 수행하기 위한 무료쏘프트웨어를 채용해 나가고 있는 오늘의 콤퓨터세계에서 하나의 작은 혁명으로 된다.

이 책에서는 Red Hat Linux 7.1 에 기초한 Linux 의 설치와 리용에 대한 소개를 주었다. Red Hat Linux 7.1 은 Linux 워크스테이션과 탁상형콤퓨터에서는 물론 봉사기에서도 쉽게 리용할수 있으므로 널리 쓰이는 Linux 의 판본이다. Linux 가 제공하는 넓은 범위의 특성과 성능을 이 책에 다 주기는 어렵다. 그러므로 여기서는 Linux 사용자가 다루고 배우는데 필요한 Linux 특히 Red Hat Linux 에 대한 지식만을 준다.

이 책의 독자대상

이 책은 실제로 PC 를 다루는 사람들을 대상으로 한다. 아직 Linux 가 Windows95/98/Me 와 Mac OS 와 같은 상품적인 탁상형조작체계에서처럼 편리한 사용환경을 제공하지는 못하지만 누구나 쓸수 있으며 Windows 를 파악할수 있는 수준의 사람이라면 능히 배울수 있다.

다시말하여 콤퓨터기초지식 즉 하드디스크와 RAM 의 차이를 식별할수 있고 비데오기판, 음성기판과 같은 주변장치의 기본구성(Windows 에서)을 리해할수 있으며 자료들이 콤퓨터에서 어떻게 구조화되고 보관되는가(실례로 등록부, 보조등록부, 파일의 구별)를 알고 있는 사람이라면 더 쉽게 배울수 있다.

이 책의 목적은 일반콤퓨터사용자들에게 Linux 를 소개하는것이다. 여기서는 사용자들이 Windows(혹은 Mac OS)와 일반 PC 하드웨어를 잘 알고 있다고 본다. 이러한 가정이 없다면 이 책은 Linux 조작체계에 대한 상세한 소개보다 콤퓨터기초개념에 대한 통론으로 될것이다.

Windows 나 Macintosh 체계를 이미 써보았지만 DOS 방식에 대한 경험이 없어도 Linux 의 세계에 들어 가기전에 지령행이 Windows 계에서보다 더 강력하고 더 무게 있

게 쓰인다는것을 알수 있다.

Windows 를 능히 다룰수 있는 사용자라면 Linux 를 다룰수 있는 준비가 되였다고 볼수 있다. 이 책을 통해서 독자들은 Linux 에서 충분히 작업할수 있게 되며 우수한 Linux 사용자로 준비할수 있게 된다.

이 책은 또한 SOHO(소규모사무/가정사무)사용자나 소규모망의 관리기에게도 해당된다. 많은 경우에 사무환경 그리고 파일봉사기, 인트라네트 Web 봉사기, 경로기로서의 기능을 Linux 가 어떻게 실현하는가를 배우게 될것이다.

요구사항

이 책을 리용하자면 최소한 다음과 같은 성능을 가진 개인용콤퓨터가 있어야 한다.

- Pentium CPU 혹은 그이상(리론적으로 386 체계에서도 Linux 를 쓸수 있지만 성능상 사용자들에게 만족을 줄수 없다.)
- 32MB 혹은 그이상의 RAM(Linux 의 성능을 충분히 발휘하려면 64MB 이상의 기억용량이 있어야 한다.)
- 적어도 1GB 의 빈 공간을 가진 하드디스크, 워크스테이션이나 무릎형콤퓨터에서는 1.5GB 의 빈 공간을 가진 하드디스크(Linux 와 Linux 응용프로그람을 실행하는데 충분한 공간용량은 3GB 이상이다.)
- CD-ROM 구동기(ATAPI/IDE CD-ROM 혹은 SCSI CD-ROM 구동기)
- 존재하는 자료나 응용프로그람을 복구해야 할 경우 현행체계의 여벌복사
- 비데오기판과 VGA 혹은 성능이 더 높은 현시장치
- 건반과 마우스

또한 다음과 같은것을 추가적으로 더 선택할수 있다.

- 인쇄기(PCL 을 위한 Opt 혹은 가능하면 PostScript 인쇄기)
- 음성기판과 스피카
- 전화모뎀, ISDN 모뎀 혹은 기타 고속련결장치

이 책의 구성

이 책은 7 개 편으로 구성되여 있다. 즉

제 1 편. Linux 안내 이 편에서는 조작체계로서의 Linux 와 GNU 일반공개사용허가(GPL:General Public License)쏘프트웨어모형의 기초로서의 Linux 를 소개한다. 그리

고 Linux 의 가장 대표적인 배포물(판본)에 대하여 설명하고 Linux 를 설치하기 위한 준비를 한다.

제 2 편. Red Hat Linux 7.1 의 설치 이 편에서는 독립적인 콤퓨터에서 Linux 체계를 설치하는 방법을 고찰한다. 여기서는 Linux 를 설치하기 위한 준비와 특수한 경우 Linux 를 설치하는 방법(CD-ROM 구동기가 없는 체계에서의 설치)에 대하여 고찰한다.

제 3 편. Red Hat Linux 7.1 에서 탁상환경의 리용 이 편에서는 GNOME(표준 Red Hat Linux GUI)탁상관리기를 가진 X Windows 를 리용하는 방법을 고찰한다. 여기서는 몇가지 일반 X Windows 응용프로그람을 개괄하여 설명한다.

제 4 편. 기본요점습득 이 편에서는 실지 Linux 를 파악하는데 필요한 본질적인 기능 즉 파일관리, 체계구성, 인쇄기와 주변장치에 대하여 설명한다.

제 5 편. 기본접속 이 편에서는 Linux 체계를 전화접속방식(dial-up)으로 인터네트에 접속시키는 방법을 설명하며 WWW(World Wide Web)와 전자우편(e-mail)의 리용방법을 설명한다.

제 6 편. 소규모사무/가정사무(SOHO)에서 Linux 의 리용 값이 눅고 강력한 콤퓨터환경을 제공하는 Linux 는 소규모사무나 가정사무에도 적합하다. 이 편에서는 Red Hat Linux 7.1 을 망사무환경에 설치하는 방법을 설명하고 Unix 와 Windows 망을 위한 파일 및 인쇄봉사기로서 Linux 를 리용하는 방법과 망환경에서 Linux 를 리용할 때 기본적인 보안문제에 대해서도 취급한다.

마지막으로 이 편에서는 Linux 에서 DOS 와 Windows 의 량립성과 통합성을 시험한다. 량립성과 통합성이란 Windows 환경에서 실행되는 쏘프트웨어 및 응용프로그람을 그대로 Linux 에 옮긴다는것을 말한다.

제 7 편. Red Hat Linux 7.1 을 Web 봉사기와 전자우편봉사기로 리용 Linux 는 SOHO 에서의 리상적인 파일봉사기와 인쇄봉사기로서만이 아니라 기업 Web 봉사기와 우편봉사기를 만드는데 리용될수 있다. 이 편에서는 Linux 에 기초한 Web 봉사기 및 우편봉사기를 설정하는 방법을 설명한다.

Linux 에 대한 일정한 경험이 있으면 제 2 편 《Red Hat Linux 7.1 의 설치》로 곧바로 넘어 가서 Linux 체계의 설치를 시작할수 있다. Linux 를 써보지 못했으면 제 1 편 《Linux 안내》로부터 시작하는것이 좋다. 콤퓨터는 작업능률을 높이는데서 추가적인 능력을 발휘하는 도구이며 Linux 는 그 역할을 강화할수 있는 훌륭한 수단이라는것을 명심해야 한다.

제 1 편. Linux 안내

제 1 장. Linux 란 무엇인가

이 책의 차례를 읽어 보았다면 Linux 에 대한 일정한 표상을 가졌을것이다. 앞부분을 읽어 보지 못했다면 한번 읽어 보는것이 좋다.

콤퓨터산업분야에서 Linux 의 개발은 참으로 놀랄만한것이다. Linux 의 개발은 고급한 쏘프트웨어라고 해도 업무적인 쏘프트웨어산업의 재정적인 지반에 반드시 의존하지 않는다는것을 보여 준다. Linux 의 력사는 무료쏘프트웨어가 어떻게 발전하여 왔으며 업무적인 판매제품들과 어떻게 당당히 경쟁하게 되였는가를 시사해 주고 있다.

Linux 는 기술적으로 뛰여 난것으로 하여 Microsoft, Apple, Novell, IBM 회사에서 만든 조작체계에 맞설수 있는 힘 있는 경쟁자로 되였다. 응용프로그람에 대한 수요가 더욱더 많아 지는것으로 해서 Linux 는 대기업들을 위한 Solaris, HP-UX, SCO-Unix 와 같은 Unix 계렬로부터 파생된 조작체계들과도 역시 경쟁자가 되였다.

이 장에서는 Linux 의 력사를 간단히 서술한 다음 여러가지 기능적인 응용프로그람을 실행하기 위한 강력한 대용조작체계로서의 Linux 가 가지는 주요특성을 설명한다. 이 장의 마감에서는 Linux 에서 리용되는 무료쏘프트웨어모형을 설명하고 그것이 쏘프트웨어산업전반에 미치는 영향을 고찰한다.

Linux 의 력사

콤퓨터분야에 Linux 가 등장함으로써 사람들은 오래동안 지배되여 오던 Unix 세계에서 벗어 나게 되였다. 조작체계로서의 Unix 는 이미 탁상형콤퓨터시대가 시작되기 퍽 오래전에 세상에 나왔다. Unix 는 소형콤퓨터(minicomputer)와 대형콤퓨터(mainframe)가 기업계의 주류를 이루던 시기인 1969 년에 개발되였다. Unix 는 오늘날 의뢰기-봉사기방식의 인트라네트망에서 흔히 쓰이고 있지만 아직도 교육분야는 물론 기업분야에서도 계속 쓰이고 있다.

Unix 의 결함은 기업소나 대학의 콤퓨터쎈터에서만 리용할수 있고 거기를 벗어 나 Unix 체계를 리용하고 싶어 하는 일반적인 프로그람작성자들과 개발자들은 리용할수 없는것이다. PC 에서도 Unix 의 판본이 오래동안 많이 리용되여 왔지만 아직도 소형콤퓨터와 대형콤퓨터, 오늘날의 봉사기들에서 쓰이는 조작체계들의 성능을 따라 가지 못하고 있다. 또한 Unix 의 업무용판본은 가격이 높았으며 지어 어떤 때는 이 가격이 Unix 를 설치하기 위해서 설계된 PC 장치보다 가격이 더 비싼 경우도 있었다. 이로부터 광범한 가동환경에서 쓸수 있는 Unix 형의 조작체계가 필요하게 되였고 그러한 의미에서 Linux 가 나오게 되였다.

80 년대 중엽에 리챠드 스톨맨과 무료쏘프트웨어협회(FSF)가 이 대용조작체계에 대한 연구를 시작하였다. 거의 10 년만에 그들은 핵심부를 제외하고 주요 Unix 부분품들을 대신할수 있는 모든 기능들을 개발하였다. 한편 핀란드 헬싱키종합대학학생이였

던 리누스 토발즈는 1991 년에 첫 Linux 핵심부를 개발하였다. 이 두개의 연구가 결합되여 오늘의 Linux 로 되였다.

토발즈는 원래 취미본위로 Linux 를 만들어 보려고 하였다. Linux 의 첫 판본은 사용자들의 마음에 들지는 못했지만 대신 Unix 프로그람작성자들로 하여금 핵심부를 만들어 볼수 있도록 기본골격의 기능을 제공하였다. 핵심부의 역할은 조작체계의 핵으로서 모든것이 원활하게 실행되도록 지원하는것이다. 즉 안정하고 강력한 핵심부가 없이는 조작체계를 만들수 없다.

프로그람작성자들은 토발즈의 핵심부를 FSF 의 연구에 통합시킴으로써 완전한 조작체계를 이루는 기본쏘프트웨어를 만들어 냈다. 이때로부터 Linux 는 정식 조작체계로 불리울수 있게 되였다. 1992 년 3 월에 핵심부 1.0 판이 Linux 의 첫 공식계렬로 발표되였다. 이때부터 X Windows 의 콤파일러로부터 망쏘프트웨어에 이르기까지 거의 모든 일반 Unix 도구들이 Linux 에서 리용되게 되였다.

Linux 는 PC 를 위한 우수한 Unix 계조작체계로 계속 발전하여 왔다. 하드웨어는 현재 가장 대중적이고 일반적인 주변장치를 폭 넓게 지원한다. 즉 Sun Microsystems 의 SPARC 체계와 같은 중간급워크스테이션의 능력을 발휘하도록 PC 에까지 이식되고 있다. 그중에서도 특히 Linux 는 PowerPC 와 Compaq Alpha 가동환경에 이식되고 있다. 비록 오늘날의 Linux 가 상표이름을 달지 못한것으로 해서 기술적으로는 Unix 만 못하지만 거의 모든 중요수법에서는 Unix 와 기능이 같다.

조작체계로서의 Linux

《Linux》란 말은 사실 좀 애매하다. 《Linux》란 말은 두가지 뜻으로 쓰이는데 좁은 의미에서는 핵심부 그자체를, 넓은 의미에서는 핵심부에서 실행되는 응용프로그람들의 묶음까지를 포괄하는데 그것을 흔히 배포물이라고 한다. 핵심부의 사명은 응용프로그람을 실행할수 있는 기본환경을 제공하는것이다. 여기에는 장치의 기본대면부와 과제관리, 현행 실행프로그람을 위한 체계가 포괄된다.

특별한 의미에서 어떤 경우에는 현행판본의 Linux 즉 핵심부의 현행개정판만이 있을뿐이다. 토발즈는 Linux 개발세계에서 핵심부만을 자기의 개발령역으로 보존하고 핵심부에서 실행되는 모든 응용프로그람과 편의프로그람은 세계의 수천명에 달하는 다른 Linux 개발자들에게 맡기고 있다.

Linux 핵심부에서 실행되는 응용프로그람들의 묶음을 일반적으로 《Linux》라고 하는데 여기에는 Linux 의 많은 판본들이 속한다. 자체의 유일한 특성을 가지는 매개의 배포물에는 각이한 설치방식과 각이한 특성들의 묶음 그리고 각이한 갱신경로가 포함된다. 그러나 모든 배포물이 근본적으로는 Linux 이기때문에 거의 모든 경우에 현행판본의 배포물에서 작업하는 응용프로그람은 현행판본의 다른 배포물에서도 작업할수 있다.

주의 제 2 장 《배포물의 선택》에서 Linux 배포물에 대하여 구체적으로 설명한다.

《Linux》란 말의 두가지 용법에서 흥미 있는것은 《조작체계》라는 말과 같은 의미에서 오는 혼동을 정확히 갈라 보는것이다. 지금까지 사용자들은 조작체계가 핵심부를 둘러 싼 응용프로그람의 큰 묶음이라는 의미로 알고 있었다. 이것이 바로 Windows 95, 98, Me, NT, 2000 그리고 Macintosh OS이다.

순수한 기술적의미에서 조작체계는 응용프로그람을 개발하는데 필요한 기본체계기능을 제공하는 훨씬 더 작은 핵심부이다.

우와 같은 두가지 리유로부터 Linux를 조작체계로 부른다. Linux 핵심부의 특성중의 하나는 탁상형 PC 우에서 실행되는 다른 조작체계와는 달리 다중과제처리와 다중사용자관리를 할수 있다는것이다.

다중과제처리용조작체계

《다중과제처리》라는 말은 그 의미를 잘 모르는 사람들속에도 널리 알려 졌다.

Windows 3.1로부터 Windows 95에로 발전하면서 유명해 진것이 다중과제처리성능이였다.

체계가 다중과제처리를 한다는것은 한번에 한개이상의 응용프로그람 혹은 프로쎄스를 실행한다는것을 의미한다. 실례로 체계는 사용자가 Word 상에서 문서를 편집하고 있어도 문서를 인쇄할수 있고 파일을 복사할수 있으며 인터네트에 접속할수도 있다. 배경에서 동작하는 과제들이 있어도 전경의 Word 프로그람이 지장을 받거나 동작하지 않는 현상은 일어 나지 않는다.

이것이 다중과제처리의 비결이다. 즉 한개의 처리기를 가지고 동시에 여러개의 과제를 수행할수 있다. 물론 하나의 CPU는 한번에 한개 지령밖에 실행할수 없다. 다시말해서 어떤 순간에 하나의 동작만 일어 날수 있다. 이러한 프로쎄스들의 요구를 받아 들여 과제들사이의 빠른 절환으로 동시적인 작용이 나타나게 하는것이 다중과제처리이다.

다중과제처리가 진행될 때 Word 편집기를 가동시키고 있는 사용자는 콤퓨터가 실행하는 다른 작업들에 대하여 느끼지 못한다. 모든 프로쎄스들이 원활하게 실행되는것처럼 보이며 콤퓨터는 이에 능란하게 응답한다.

력사적으로 Unix 체계는 다중과제처리에서 Windows 보다 훨씬 더 우월했다. Unix는 큰 기관의 봉사기들과 고성능워크스테이션에 해당되는것으로서 동시에 많은 응용프로그람들을 실행할수 있게 설계되였다. 오늘날에는 다중과제처리를 실행하는 Windows 2000과 그에 선행한 Windows NT 4.0만이 환영을 받고 있다. Windows 95/98/Me는 큰 규모의 동시처리를 효과적으로 조종하는데서 일부 혼란을 가져 오기때문에 그의 다중과제처리능력을 아무리 선전해도 환영을 받지 못한다.

앞으로는 2중 Pentium Ⅲ과 같은 다중처리기를 소유한 콤퓨터들도 제공된다. 이러한 체계에서는 실제상 같은 시간에 정확히 두가지 동작을 수행할수 있다. 다중처리기에 다중과제처리기능까지 결합하면 콤퓨터가 동시에 원활하게 실행할수 있는 응용프로그람들의 수가 크게 증가한다.

다중사용자조작체계

다중과제처리용조작체계보다 더 중요하게 보는것은 Linux(Unix 와 Unix 계의 모든 판본과 같이)가 다중사용자조작체계라는것이다.

Windows 의 모든 판본과 Mac OS 는 단일사용자체계이다. 다시말하여 이러한 조작체계에서는 어떤 순간에 한명의 사용자만이 가입해서 응용프로그람을 실행할수 있다. 반면에 Linux 를 리용하면 여러명의 사용자들이 동시에 작업할수 있으며 조작체계의 다중과제처리능력도 완전히 제공된다. 기본우점은 Linux 를 응용프로그람의 봉사기로도 리용할수 있는것이다. 탁상형콤퓨터나 말단으로부터 LAN 을 거쳐 Linux 봉사기에 가입한 사용자들은 실제상 자기의 탁상 PC 만이 아니라 봉사기에 있는 응용프로그람도 리용할수 있다.

Linux 응용프로그람

Linux 는 조작체계로서 거의 모든 형태의 응용프로그람을 개발하는데 리용될수 있다. Linux 에 쓰이는 응용프로그람에는 다음과 같은것들이 있다.

본문 및 단어편집프로그람 Linux 는 WordPerfect, StarOffice, Applixware 와 같은 업무적인 문서처리쏘프트웨어에 보충적으로 문서파일을 편집하고 자동적으로 문서를 처리하는 강력한 도구를 제공한다.

프로그람작성언어 폭 넓고 다양한 프로그람작성언어 및 스크립트언어와 도구들은 Linux 와 모든 Unix 조작체계에서도 리용할수 있다. 이 풍부한 프로그람개발도구들이 있으면 Linux 만이 아닌 Unix 와 Unix 계렬의 조작체계에서 실행되는 새로운 응용프로그람들을 쉽게 개발할수 있다.

X Windows X Windows 는 Unix 의 도형사용자대면부(GUI)이다. X Windows 는 거의 모든 Unix 체계에서뿐아니라 Linux 에서도 유연하게 실행되도록 구성할수 있는 GUI 환경이다. X Windows 에서 실행되는 많은 응용프로그람들은 Linux 를 리용하기 쉬운 조작체계로 되게 한다.

주의 제 6 장에서부터 12 장까지에서 X Windows 의 완전한 적용범위를 설명한다.

인터네트도구 Linux 는 널리 알려 진 Netscape 와 Mosaic 와 같은 쏘프트웨어뿐만아니라 보다 많은 인터네트쏘프트웨어를 제공한다. 이것은 문자방식과 도형방식의 전자우편응용프로그람, 인터네트봉사기(Web 봉사기, 우편봉사기, 새소식봉사기)를 만드는데 필요한 쏘프트웨어, 국부망이나 모뎀을 통하여 인터네트에 접속하기 위한 완전한 망지원 등을 포괄한다.

자료기지 Linux 는 의뢰기-봉사기형자료기지응용프로그람을 실행할수 있는 가동환경을 제공하는데 이 환경들은 Unix 의 모든 가동환경들과 비슷하다. mSQL, PostgreSQL 과 같은 강력한 자료기지들은 처음부터 Linux 에서 리용되였다. 특히 집단적인 정보체계에서 Linux 의 인기가 올라 감에 따라 Linux 를 위한 업무용관계형자

료기지봉사기들의 수가 증가하였다. 오늘날 Oracle, Sybase, Informix 는 모두 Linux 에 대한 관계형자료기지제품을 제공한다.

DOS 와 Windows 호환쏘프트웨어 제 30 장 《Red Hat Linux 7.1 과 DOS/Windows》에서 보게 되겠지만 Linux 는 높은 준위의 안전성과 호환성을 가지고 DOS 쏘프트웨어를 실행할수 있으며 Windows 쏘프트웨어를 실행하기 위한 여러개의 도구들도 제공한다. 실제로 이 책은 Linux 에서 Windows 의 Microsoft Word 를 리용하여 작성되였다. 이것은 Windows 환경에서 작업하려는 Linux 의 능력에 대한 강력한 증거물로 된다. 또한 모방기들은 다른 일반적인 콤퓨터체계(Macintosh 및 Atari ST 콤퓨터 등)에서도 리용할수 있다.

이러한것은 극히 일부에 지나지 않는다. Linux 에는 더 많은 응용프로그람들이 존재한다. Linux 쏘프트웨어를 찾을수 있는 원천은 Linux Software Map 인데 이것은 WWW 의 http://www.execpc.com/lsm/ 에서 찾을수 있다.

무료쏘프트웨어로서의 Linux

Linux 가 기대하는 조작체계로서의 모든 성능을 갖추자면 비용이 많이 들것이라고 생각할수 있다. 그러나 이와는 반대로 Linux 핵심부와 거의 모든 Linux 응용프로그람은 쏘프트웨어의 복사와 재배포에 대한 제한이 없이 인터네트에서 무료로 리용할수 있다.

우선 Linux 핵심부는 GNU 일반공개사용허가(GPL)밑에서 배포된다. 스톨맨무료쏘프트웨어협회가 추가하는 이 특수한 쏘프트웨어사용허가는 공개배포물과 보다 중요하게는 쏘프트웨어의 공개개발을 촉진시킨다. 업무용쏘프트웨어에 대한 일반적인 쏘프트웨어사용허가와는 달리 GNU 사용허가는 누구나 쏘프트웨어를 재배포할수 있게 하며 지어는 보수를 바라고 재배포를 한다 해도 여전히 실시될것이다. 다시말하여 누구나 GNU 쏘프트웨어를 가질수 있고 자기들이 바라는대로 변경시킬수 있으며 그것을 재배포할수 있지만 자기들로부터 사용허가된 GNU 쏘프트웨어를 산 사람들이 그것을 돌리고 다시 재배포하는것은 막을수 없다.

대부분 Linux 는 GNU 일반공개사용허가밑에서 리용된다. 그러므로 대다수의 제작자들은 Linux 의 무료배포물과 업무용배포물을 다 리용할수 있다.

무료쏘프트웨어에서 취급법은 공개령역쏘프트웨어에서와 꼭 같지 않다. GNU 제품들에서 쏘프트웨어개발자들은 그 쏘프트웨어에 대한 저작권을 가진다. 그리고 GNU 사용허가밑에서 그 쏘프트웨어에 대한 배포를 중지시킬수 있다. GNU 사용허가에서 특징적인것은 많은 사람들이 응용프로그람을 부단히 개발할것을 요구하는것이다. 매 사용자는 자기들이 중요하거나 필요하다고 간주하는것을 갱신한 다음 그 쏘프트웨어를 재배포할수 있다.

이러한 프로쎄스는 사용허가된 모든 GNU 쏘프트웨어가 자기의 완전한 원천코드와 함께 배포되여야만 가능하다. 원천코드를 리용할수 없고 따라서 변경할수도 없는 업무용쏘프트웨어와는 달리 GNU 쏘프트웨어는 변경할수 있고 전용으로 만들수 있을뿐아니라 흥미를 가지고 능력이 있는 사용자들이 그렇게 해보도록 한다.

사실상 이러한것으로 해서 Linux 와 Linux 응용프로그람이 성과적으로 개발되게

되였다. Netscape 는 자기의 Web 열람기를 개발하는데 이것을 리용하였다. Netscape 가 있으므로 해서 누구나 기초적인 GNU 원리들을 리용하면서 Web 열람기를 무료로 리용할수 있고 Web 열람기원천코드를 리용할수도 있으며 그것들을 재배포할수도 있다.

Linux 를 위한 업무용응용프로그람

이 책에서 후에 보게 되지만 Linux 에는 업무용배포물뿐아니라 업무용응용프로그람들도 있다. 이러한 경우에 대부분의 제품들은 Linux 세계의 GNU 표준보다 더 제한된 합의밑에서 사용허가된다.

Linux 의 배포물이 자유롭게 재배포될수 없는 업무용구성요소들을 포함한다고 해도 이것은 모든 Linux 배포물에서 볼수 있는 Linux 핵심부와 핵심응용프로그람들에 적용되는 우선적인 GNU 사용허가를 변화시키지 못한다. 만일 임의의 응용프로그람에 대한 초기사용허가가 GNU 일반공개사용허가라면 재배포된 쏘프트웨어복사물의 사용허가도 GNU 일반공개사용허가로 된다.

▶미리 알아두기

이 장에서는 Linux 세계에로의 첫 발을 들여 놓은 셈이다. 기본적인 Linux 의 구성요소와 원리가 무엇인지 알았고 Linux 의 우수한 특성들로 해서 Linux 에서 많은 응용프로그람들도 리용할수 있다는것을 알게 되였다.

제 2 장에서는 Linux 배포물의 원리와 현재 많이 쓰이는 배포물들에 대하여 그 의미를 시험해 본다.

제 3 장에서는 PC 에 Linux 를 설치하기 위한 실천적인 준비단계들을 고찰한다.

제 4 장과 제 5 장에서는 여러가지 설치방법과 그와 관련된 내용, 가능성을 고찰한다. 다음 제 6 장에서부터 제 12 장까지에서는 Linux 탁상환경을 설명한다.

제 4 장과 제 5 장에서는 개별적인 PC 에 Linux 를 설치하는 방법을 본다. 그리고 CD-ROM 구동기가 없는 PC 에서 Linux 를 설치하는 특수한 방법도 고찰한다. 그러므로 이 책에서는 5 개의 부분으로 갈라서 Linux 의 리용방법을 고찰한다. 이 책은 X Windows 와 GNOME, 기본적인 Linux 도구, 인터네트에 Linux 를 접속하기, 소규모사무 혹은 가정사무(SOHO 환경)에서 Linux 의 리용 등을 포괄한다.

제 2 장. 배포물의 선택

제 1 장 《Linux 란 무엇인가》에서 본것처럼 Linux 를 개발하는 총적인 수법은 전통적인 업무용쏘프트웨어개발수법과는 다르다. 완전한 Linux 체계를 이루는 대부분의 구성요소 즉 핵심부(조작체계의 심장부), 장치모듈 및 여러가지 응용프로그람과 편의프로그람은 세계의 각 곳에 분산되여 있는 개발자들로 이루어 진 집단이 만든다.

이러한 구성요소들은 대부분이 GNU 일반공개사용허가(부록 2 를 참고)와 같은 무료재배포를 승인하는 사용허가밑에서 배포된다.

이러한것들로 해서 Linux 사용자들은 작업중인 Linux 체계를 어떻게 한데 모아두며 그 체계에 포함되는 구성요소가 무엇인가 하는 의문을 가지게 된다.

이러한 곤경에서 벗어 나기 위해 Linux 배포물형식을 취하였다. 배포물은 설치할 수 있게 준비된 Linux 체계를 미리 묶어 놓은것이다. 이것들은 무료로 리용할수 있는 것으로부터 완전한 업무용에 이르기까지의 여러가지 형태로 존재하며 각이한 주요응용프로그람, 편의프로그람, Linux를 쉽게 리용할수 있게 하는 관리도구를 제공해 준다.

이 장에서는 몇가지 중요하면서도 널리 알려 진 배포물들을 개괄적으로 고찰한 다음 이 책의 CD-ROM 에 들어 있는 배포물인 Red Hat Linux 7.1 의 최신판에 대하여 설명한다.

배포물이란 무엇인가

Widnows 98 과 2000, Mac OS, Solaris 와 HP-UX 와 같은 업무용 Unix 체계를 비롯한 업무용조작체계의 세계에서는 배포물에 대한 개념을 리해하기가 좀 어렵다.

이 모든 경우에 조작체계의 이름은 매우 특수한 제품을 의미한다. 실례로 Windows 98 은 Microsoft 회사가 제공하는 Windows 편의프로그람, 응용프로그람, 구동프로그람으로 이루어 지는 완전한 묶음이다. 그러므로 변화시켜 볼 여지가 없다. 사용자들이 자기들의 체계에 어떤 응용프로그람이나 구동프로그람, 편의프로그람을 추가해도 Windows 98 은 이것을 자기의 부분으로 간주하지 않으며 자기의 완전한 묶음으로서의 쏘프트웨어와 도구외에는 다른 제품을 기술적으로 존속시키지 않는다.

그러나 Linux 세계에서는 이러한 정의를 무시한다. 엄밀한 의미에서 말할 때 《Linux》는 바로 핵심부(조작체계의 심장부)이지만 다른 한편 《Linux》라는 말에는 핵심부로부터 시작하여 핵심부에 한데 묶여 있으면서 Linux를 실행하는 기능적인 체계를 이루는 Linux에 기초한 응용프로그람들의 집합에 이르기까지의 모든것이 속할수 있다.

지정할수 있는 응용프로그람, 편의프로그람, 구동프로그람 및 핵심부의 명백한 묶음들에 일련의 부족점이 있게 되는데 그것은 Linux 가 각이한 사용자들의 요구에 따라 그들에게 문을 열어 놓았기때문이다. 이것이 바로 배포물로 나타난다.

배포물은 각이한 판본의 핵심부에서 만들어 지고 각이한 묶음의 응용프로그람, 편의프로그람, 도구, 구동프로그람모듈을 포괄하고 있으며 각이한 설치프로그람을 제공하고 체계관리를 쉽게 하기 위하여 프로그람들을 갱신할수도 있다.

이러한 유연성이 무질서를 초래할수 있는것처럼 생각할수도 있다. 이것은 리론적인 결론이다. 어떻게 Linux 를 끊임없이 변경시키고 어떻게 임의의 체계들에 Linux 응용프로그람을 설치하고 실행할수 있는 몇가지 준위의 합리적인 담보를 가능하게 하겠는가?

다행히도 이 체계는 작업하기 쉽다. 서로 다른 Linux 판본의 기초에는 거의 모든 Linux 체계들에서 리용할수 있는 응용프로그람들을 개발하는데 필요한 호환성을 제공하는 스레드(thread)의 류사성이 놓여 있다.

대부분의 Linux 배포물에는 응용프로그람개발자들이 임의의 Linux 체계에서 볼수 있는 기초프로그람과 편의프로그람, 서고의 공동묶음이 들어 있다. 또한 대부분의 현행 Linux 배포물들은 Linux 파일체계계층표준(LFHS)과 같은 표준제품을 보존하고 있다.

그러므로 Netscape 나 Corel WordPerfect 와 같은 대규모업무용응용프로그람들을 Linux 용으로 개발할수 있고 대부분의 Linux 체계에서 리용할수 있다. 또한 응용프로그람이 요구하는 프로그람이나 서고와 같은 구성요소를 갖추고 있지 못했다고 해도 그

것을 쓸수 있도록 인터네트에 있는 큰 규모의 Linux 쏘프트웨어저장고로부터 그것들을 내리적재할수 있다.

주요배포물에 대하여

대부분의 Linux 배포물은 무료로 리용할수 있다. 그것들은 인터네트에서 Metalab FTP 싸이트나 MIT 에 있는 TSX-11Linux 저장고와 같은 Linux 쏘프트웨어서고에서 내리적재할수 있다.

일러두기 배포물을 제공하는 Linux 싸이트의 목록과 CD-ROM 제작자들의 목록은 부록 2 《Linux 정보원천》에서 볼수 있다.

물론 Linux 배포물의 크기가 평균 수십 MB 로부터 수 GB 까지의 아주 큰 규모일수도 있으므로 대다수의 사용자들은 완전한 Linux 배포물을 내리적재하는데 드는 값 비싼 직결시간과 기억공간을 소비하려고 하지 않는다. 이러한 상황을 고려하여 대부분의 Linux 배포물은 그것을 만드는 조직 혹은 추가적인 Linux 쏘프트웨어와 문서를 포괄하고 있는 하나이상의 배포물이 들어 있는 CD-ROM 을 제공하는데 이 CD-ROM 은 제 3 자의 원천들로부터 제공 받을수 있다.

실례로 CheapBytes 의 MONDO Pack(www.cheapbytes.com)27 판본은 11 장의 CD-ROM 으로 되여 있으며 여기에는 최신판의 Red Hat, Slackware, Mandrake, StormLinux 배포물이 들어 있다. 또한 LinuxMall(www.Linuxmall.com)은 Linux MegaPak 2.0 을 제공한다. 이것은 MONDO Pak 와 같은 배포물외에도 최신판의 Debian, Storm, S.u.S.E., FreeBSD TurboLinux 배포물이 들어 있는 16 장의 CD-ROM 으로 된 판본이다.

주의 다른 Linux 배포물들과 마찬가지로 FreeBSD 도 Unix 의 일종이다.

일반적으로 CD-ROM 은 2～50 딸라범위에서 무료 Linux 배포물로 복사되는데 Linux Mega Pak 와 같은 대중용배포물은 30 딸라미만이다.

주요배포물

수많은 Linux 배포물들이 있다. 최근년간에 Red Hat, Slackware, Caldera, S.u.S.E.,Debian, Corel 의 6 개 배포물들이 가장 대표적인것으로 등장하였다. 일반적으로 이러한 배포물들은 Linux 세계에서 가장 오랜 력사를 가지고 있으며 이것들은 함께 시장공유를 조종한다. 뿐만아니라 Linux 배포물을 포함하는 다른 배포물들과 판매제품의 기초로 되는 이 6 개의 배포물들은 모두 무료로 리용할수 있다.

이 6 개를 제외하고도 Yggdrasil 과 같은 오랜 배포물들과 TurboLinux 와 easyLinux 와 같은 새로운 배포물을 포함한 2 부류의 Linux 배포물이 있다.

Red Hat

Red Hat Software(www.redhat.com)가 만든 Red Hat Linux 는 많은 사용자들속에서 인기 있는 Linux 배포물이다. 이 배포물은 많은 업무용 Linux 쏘프트웨어개발자들을 대상으로 한 기초적인 배포물이며 많은 Linux 배포물들의 성능을 검사하는 기준이다.

Red Hat 는 조작체계의 설치와 갱신을 위한 도구와 쏘프트웨어판매제품의 설치와 해제, 추적할수 있는 도구들이 잘 설계된것으로 하여 명성을 떨쳤다.

Red Hat Linux 는 2000 년에 있은 Network Magazine 의 년간제품판정을 비롯하여 많은 판정들에서도 우승하였다. 1998 년에 InfoWorld 의 Web 싸이트손님들을 통하여 년간 제품구입에 대한 수요를 조사해 본 결과 Red Hat Linux 가 단연 첫 자리를 차지하였다(Red Hat 는 27%의 득점표를 받았고 OS/2 Warp4 는 8%의 득점표를 받았다). Red Hat 는 또한 영화《타이태니크》의 동화편집작업을 비롯해서 Linux 의 업무적인 생활력을 증명하는 대상과제들에 리용되기도 하였다.

Red Hat Linux 는 인터네트의 공개 Linux 서고들에서 내리적재할수 있는 무료판본으로 리용할수 있고 설명서와 CD-ROM, 여러개의 무료쏘프트웨어로 제공되는 업무용응용프로그람을 비롯하여 40 딸라 업무용판본으로도 리용할수 있다. Red Hat Linux 7.1 은 또한 Deluxe Workstation 및 Professional Server 편집에 리용될수 있다.

Red Hat Linux 7.1 은 현행배포판본이며 이 책에서 설명된다. 이 장에서는 Red Hat 에 대하여 더 심도 있게 고찰한다.

주의 CD-ROM 의 Red Hat Linux 7.1 배포물외에 ftp.redhat.com 에 있는 Red Hat FTP 봉사기 혹은 ibiblio.org/pub/Linux/distributions/redhat 에 있는 ibiblioLinux 서고로부터 최신판을 항상 내리적재할수 있다. 이 싸이트들을 쉽게 호출할수 없으면 http://www.redhat.com/download/mirror.html 로부터 제공되는 배포물의 복사판인 수많은《반사》싸이트들을 리용할수 있다.

Slackware

Red Hat Linux 가 명성을 날리기전에는 Slackware 가 유명했다. Slackware 는 아직도 대중적인 배포물로 되고 있지만 최근에 있은 기업정비로 하여 자기의 특색을 살리지 못하고 있다. 패터릭 볼커딩은 Slackware 대상과제의 책임자이며 현재 자기의 개발자들을 위한 다른 후원자를 찾고 있다. 적재싸이트나 판매를 통한 정보를 비롯하여 Slackware 에서 최근에 개발한것들에 대한 상세한 정보를 보려면 홈페지 www.slackware,com 를 참고하여야 한다.

Slackware 의 현행판본은 Slackware 7.1 이다. 이 배포물은 X Windows 를 비롯하여 넓은 범위에서 기대되는 편의프로그람, 도구, 응용프로그람과 GNU C 콤파일러, PPP 지원, 완전한 Java 지원, Java SDK(Developer's Kit)와 같은 Linux 용개발도구들을 제공한다. 대부분의 Linux 배포물들과 마찬가지로 Slackware 는 인트라네트나 인터네트의 Web 싸이트를 설정하는데 Linux 를 리용하는 Apache Web 봉사기를 제공하며 무료로 리용할수 있는 Web 열람기들도 제공한다.

이 책에 쓴바와 같이 Slackware 는 http://www.slackware.com/getslack에 있는 수많은 FTP 와 HTTP 싸이트들로부터 내리적재할수 있다.

Caldera OpenLinux

Caldera 는 지원가능한 업무용 Linux 에 대한 긴급요구에 따라 Linux 와 매우 넓은 콤퓨터시장의 두 측면을 분발시켰다. 지금은 다른 Linux 제작자들 특히 Red Hat 가 그 자리를 차지하고 있다. 그들의 의도는 검사되고 안정하며 지원기능을 가진 Linux 의 판본들을 제공하는것인데 이것은 개발과정에 안전한 Linux 배포물을 요구하는 상업시장과 응용프로그람개발자들을 끌기 위한것이다.

이 전략이 어느 정도까지 실현되였다. Linux 를 위한 Corel 의 WordPerfect 6 은 초기에 Caldera 의 망탁상배포를 목표로 하였다. Linux 를 위한 StarOffice 의 이전판본은 Caldera 배포를 지향했고 Netscape 의 FastTrack Web 봉사기프로그람에 대한 Linux 판본은 Caldera 로 옮겨 졌으며 eDesktop 과 eServer 프로그람의 부분으로 리용할수 있다.

현행 Caldera OpenLinux 배포물은 2 개의 판본 즉 워크스테이션을 위한 eDesktop 2.4 와 봉사기를 위한 eServer 2.3 으로 구성된다. 그것은 Web 으로부터 구입하거나 내리적재할수 있다. 두 배포물은 다음의 특성을 가진다.

- K 탁상환경(제 11 장 《KDE 의 리용》에서 설명된다.)
- Linux 를 위한 StarOffice 의 비업무용사용허가
- Netware 지원
- DOS 호환성에 대한 DR-DOS 의 사용허가
- Linux 열람기를 통해 볼수 있는 강력하면서도 전용화할수 있는 Linux 관리체계인 Webmin
- Linux 를 설치하기전에 사용자들이 자기들의 장치구성에 대한 정보를 제공하는 통합판본인 Partition Magic

Caldera 의 eDesktop 2.4 는 2000 년 10 월 CNet Editor 가 주관하는 제품선정대회에서 우승하였다. eDesktop 이 우승함으로써 CNet 의 편집자들은 인터네트에 기초한 관리도구에 추가된 기능들에 주목을 돌리게 되였다. Caldera 의 eDesktop 과 eServer 는 Caldera 의 Web 싸이트인 www.caldera.com으로부터 무료로 제공 받을수 있다.

S.u.S.E.Linux

S.u.S.E.Linux 는 주로 유럽에서 리용되는 공개 Linux 배포물이며 영어와 도이취어판본으로 제공된다.

이 책을 집필할 당시 S.u.S.E 의 판본은 7.1 이였으며 이것은 다른 주요배포물들과 구별되는 몇가지 특성을 가진다. S.u.S.E.는 다음의 특성을 제공한다.

- KDE 2.0 은 Xfree86 X Windows 봉사기의 최신판본이다(일부 배포물들은 한단계 낮은 판본이다.).
- $YaST_1$ 와 $YaST_2$ 이라고 하는 체계관리도구들은 전화회선을 통한 인터네트접속으로부터 스캐너와 망기판에 이르기까지의 모든것을 편리하게 구성하는데 리용된다.
- SaXz 는 X Windows 를 구성하기 위한 전용편의프로그람이다.

- StarOffice 최신판본을 복사한다.

Caldera 나 Red Hat 와 마찬가지로 S.u.S.E. 7.1 도 현재 두개 판본으로 리용된다. 하나는 S.u.S.E.Linux 7.1 일반용판본인데 여기에는 700 개이상의 프로그람들과 응용프로그람 및 구성에 대한 사용설명서가 있다. 다른 하나는 S.u.S.E.Linux 7.1 전문용판본인데 7 장의 CD-ROM 과 한장의 DVD 에 2500 개이상의 응용프로그람이 들어있는것으로서 오늘날 가장 종합적인 Linux 배포물로 되고 있다.

S.u.S.E.Linux 는 Digital Alpha, Power PC, IBM S/390 기업봉사기를 비롯하여 인텔계가동환경이 아닌 판본들도 지원한다.

Debian/GNU

6 개의 주요 Linux 제품중의 하나인 Debian/GNU(www.debian.com)는 어떤 특정한 회사가 개발한것이 아니다. Red Hat 는 Red Hat Software 에 의하여, Slackware 는 Walnut Creck 에 의하여 개발되지만 Debian/GNU 는 Linux 개발 그자체의 특성에 맞게 지원자들이 개발한것이다.

Debian 은 자체의 프로그람관리체계를 리용하는 3950 개이상의 쏘프트웨어제품을 제공한다. 이 체계는 Red Hat 배포물이 제공하는 류사한 기능들도 가지도록 설계되였다. Debian/GNU Linux 의 현행판본은 2.2.r3 이다.

Debian 배포물은 그 수법상에서 독특한데가 있다. 그것의 Web 싸이트는 무료쏘프트웨어세계에 제공한 코드를 돌려 줄것을 요구한다. 그들은 자기들의 결함을 공개하고 그 결함들을 찾기 쉽게 해주며 Debian 의 무료쏘프트웨어의 정의에 맞지 않는 응용프로그람들을 자기들의 배포물에 포함시키지 않는다(여기에는 정당한 무료재배포, 리용가능한 원천코드, 갱신들에 대한 승인과 파생작업이 포함된다.).

Corel

주요배포물들가운데서 가장 새로운것은 Corel Linux 이다. 이것은 탁상환경에서 Microsoft Windows 를 대신하는 뚜렷한 목적을 가지고 개발된 첫 Linux 배포물이다. 제 2 의 CorelLinux 판본은 다음의 특성들을 가진다.

- KDE Control Center 에 기초한 체계관리도구인 Corel Centrol Center
- 4 단계의 설치과정
- 대등한 Microsoft Windows 도구들에 명백히 기초하고 있는 User Manager 및 Print Wizard 와 같은 도구
- 현행 Microsoft Windows 콤퓨터들로부터 륜곽들과 설정들을 받아 들이는 SmartMove

Corel Linux 는 Debian 배포물에서 만들었다. 다시말하여 Debian Linux 의 특성들을 추가하여 자기들의 배포물로 판매한다. 이것은 부록 4 에 있는 일반공개사용허가밑에서 명백히 허용된다.

기타 배포물

기타 Linux 의 영어판 배포물에는 주목할만한 LinuxPro, LinuxWare, TurboLinux, Yggdrasil 들이 포함된다. 또한 부록 《세계의 모든 Linux》에는 영어판이 아닌 배포물 특히 프랑스어와 도이취어로 된 배포물들이 있다.

Linux 직결영어판배포물페지인 www.Linux. org/dist/english.html로부터 기타 배포물에 대해서 배울수 있고 련관된 Web싸이트에로 런결하는 방법을 알게 된다.

Red Hat Linux

이 책에서는 Red Hat Linux 7.1 을 취급한다. 이미 언급된바와 같이 Red Hat Linux는 현재 가장 많이 알려 지고 널리 쓰이는 Linux배포물중의 하나이다.

Red Hat Linux 가 널리 알려 지게 된것은 Red Hat Package Manager(rpm)라고 하는 프로그람관리체계때문이다. 이 체계가 있음으로 해서 쏘프트웨어응용프로그람들을 검사하고 구성할수 있게 되였으며 Red Hat Linux 의 실행준비상태도 제공할수 있게 되였다. 이것은 S.u.S.E.와 Caldera 를 비롯한 수많은 다른 배포물들에도 적용되였다. 자체의 특수한 설치프로그람과 함께 배포되거나 Red Hat(혹은 류사한 Debian)프로그람관리체계를 리용하지 않은 쏘프트웨어들처럼 불편하게 다른 프로그람묶음을 구성하지 않고도 간단한 프로그람관리도구를 리용하여 새로운 프로그람들을 내리적재하고 설치하고 실행할수 있다.

물론 프로그람관리하나만 가지고 Red Hat Linux 의 성공을 다 설명할수는 없다. 결국 Red Hat 프로그람관리체계를 실행해야 하는 주요쏘프트웨어는 Red Hat 에 의해서 무료로 리용할수 있게 만들어 졌으며 다른 배포물도 이 체계를 리용하고 있다.

프로그람관리에 의한 한가지 우점은 갱신가능성이다. 이것으로 하여 Linux 를 재설치하지 않고 Red Hat의 판본을 갱신할수 있다. 이 문제는 Slackware가 배포의 앞선 자리를 차지하고 있을 때부터 제기된것이며 현재까지도 어떤 배포물은 해결하지 못하고 있다.

Red Hat 배포물의 다른 주요특성은 이 배포물을 IntelPC 가동환경에서뿐아니라 Compaq Alpha 형콤퓨터가동환경에서도 리용할수 있다는것이다. Compaq Alpha 가동환경은 성능상 전형적인 IntelPC 보다 더 높지만 일반적으로 값 비싼 Unix 의 업무용판본을 리용한다. Red Hat 가 있으므로 이 두 체계에서 Linux 를 실행할수 있으며 배포물이 기본적으로 같기때문에 어느 한쪽의 가동환경에서 Red Hat Linux 실행체계의 관리와 구성은 쏘프트웨어의 부리우기처럼 간단해 진다.

Red Hat Linux 7.1 에서 새로운것은 무엇인가?

Red Hat Linux 7.1 은 이전의 판본 및 기타 모든 Linux 배포물들에서 찾아 볼수 없는 몇가지 새로운 특징들을 제공한다. 이 특징들은 다음과 같다.

- 2.4Linux 핵심부
- 인쇄기와 방화벽을 위한 새로운 구성도구

- XFree86 판본 4.0.3, KDE 2.1, GNOME 1.2 를 포함하는 갱신된 X Windows 체계

Linux 2.4 핵심부

지난해 Linux 핵심부의 발전이 인상적이다. Linux 핵심부에 대한 갱신은 매해 조금씩 진척되였는데 지난해 핵심부의 발전은 탁상형콤퓨터와 봉사기 그리고 손바닥형콤퓨터와 같은 작은 장치에 대하여 새로운 특성을 나타내였다. 이러한 새로운 특성은 다음과 같다.

통합 PC 하드웨어지원 2.4 판본에서는 USB 지원기능이 개선되였을뿐아니라 핵심부에 해당 구동프로그람들을 직접 통합시킴으로써 PCMCIA(PC 기판)와 PnP(Plug-and-Play) ISA 장치들의 성능이 개선되였다.

가동환경의 다양성 새로운 핵심부가 있으면 Alpha, Sparc, MIPS, PowerPC 및 ARM 콤퓨터를 비롯한 다양한 비인텔가동환경에서 Linux 를 실행할수 있다. 지어 IBM S/390 과 같은 대형콤퓨터도 지원한다.

작은장치의 지원 새로운 핵심부는 손바닥형콤퓨터, 다른 휴대용장치와 같은 소형장치들에서도 리용할수 있게 설계되였다.

매우 적은 제한 새로운 핵심부는 대량의 RAM(64GB), 가상적으로 제한이 없는 다중과제처리 및 대용량파일용다중디스크에 단일기록집(volume)을 만들수 있는 능력을 지원한다.

새로운 구성도구

지금까지 Linux 는 Red Hat Linux 체계를 구성하고 관리하기 위하여 지령행과 도형방식도구 두가지를 다같이 강력하게 지원하였다. 판본 5 부터는 이름, 전화번호와 같은 사용자정보를 설정하기 위한 사용자정보도구, 사용자의 통과암호를 쉽게 바꿀수 있게 하는 사용자통과암호도구, 도형대면부에서 파일체계를 설치하고 해제하는 파일체계도구 등도 포함하도록 확장되였다.

Red Hat Linux 는 또한 X Windows, XFree86 의 무료판본구성을 위한 도구도 제공하는데 이것을 Xconfigurator 라고 한다. Xconfigurator 는 XFree86 에 포함되여 있는 구성프로그람보다 더 리용하기 쉽다. Red Hat Linux 에서 Xconfigurator 는 비데오기판과 그의 특성에 대한 자동검출과 같은 추가적인 개선들도 제공한다.

Red Hat Linux 는 printconf 와 gnome-lokkit 라고 하는 인쇄기와 방화벽을 구성하기 위한 새로운 도구들도 포괄하고 있다. Printconf 가 있으면 표준도형사용자대면부를 통하여 국부적으로 혹은 원격으로 인쇄기들을 설정할수 있다. gnome-lokkit 는 침입자들로부터 콤퓨터와 망을 보호하기 위한 방화벽을 구성할수 있게 하며 통신요구를 결정하는 간단한 질문에 기초하여 방화벽의 규칙들을 구성할수 있게 한다.

대체로 이러한것들이 개선되면 대다수의 사용자들은 Red Hat Linux 를 보다 쉽게 관리하고 Windows 2000 과 같은 경쟁적인 체계들에서 보게 되는 도형관리도구들의 류

형을 Linux세계에로 옮길수 있게 된다.

개선된 X Windows체계

Red Hat Linux 7.1 은 Linux X Windows 체계의 최신판본인 XFree86 판본 4.0.3 을 포괄하고 있다. 이것은 보다 안정한 가동환경과 Red Hat Linux 7.1 에 있는 두개의 주요탁상환경들인 GNOME 과 KDE 를 위한 개선된 3-D 도형을 만든다.

GNOME 과 KDE 는 응용프로그람개발을 보다 쉽게 하도록 설계된 X Windows 의 탁상환경이다. 또한 보다 조화가 잘되고 전문가수준인 탁상환경을 Linux 사용자들이 만들수 있게 한다. GNOME 1.2 와 그의 짝패인 Sawfish 창문관리기는 제 10 장 《고급한 GNOME 구성》에서, KDE 2.1 은 제 11 장 《KDE 의 리용》에서 구체적으로 보게 된다.

개선된 성능특성

Red Hat Linux 7.1 은 Red Hat Linux 체계의 성능과 안정성을 개선하기 위해 장치적으로 리용할수 있는 여러가지 새롭거나 개선된 특성들을 제공한다. 이 특성들은 대칭다중처리(SMP)를 더 잘 실현하도록 한다. SMP 가 있으면 Red Hat Linux 는 다중처리기체계가 제공하는 성능을 증대시킬수 있다. 그밖에도 Red Hat Linux 7.1 은 Disk Druid 의 판본을 개선하였는데 이 판본이 있으면 설치작업을 하는 경우 구획을 설정하여 오유를 검사할수 있다.

▶미리 알아두기

지금 시점에서 Linux 란 무엇이고 각이한 배포물에 어떤 선택항목이 있는가에 대해서 어느 정도 알게 되였고 실지로 Linux 에서 작업할수 있는 준비가 되였다고 볼수 있다(Red Hat Linux 7.1의 출판판본은 책과 함께 제공된 CD-ROM에 들어 있다.).

제 3 장 《Red Hat Linux 7.1 의 설치준비》에서는 Linux 를 설치할수 있는 예비지식을 준다. 여기에는 Linux 체계를 실행하고 하드웨어를 리해하고 설치프로그람을 제공하는데 필요한 모든 정보들이 포함되여 있다.

제 4 장 《Red Hat Linux 7.1 의 설치》에서는 Red Hat Linux 체계를 설치하기 위한 실제적인 단계들을 주었다. 지정된 단계들을 거치는 과정에 Red Hat Linux 의 설치로부터 임의의 Linux배포물을 설치하는 방법들에 대하여 배우게 된다.

제 2 편. Red Hat Linux 7.1 의 설치

제 3 장. Red Hat Linux 7.1 의 설치준비

이 장에서는 Red Hat Linux 7.1 의 설치작업에 대하여 설명한다. 이 장의 대부분 내용들은 실제적인 설치과정 그자체보다도 설치과정에 영향을 주는 요인들과 관련되여 있는데 이에 대한 고찰은 Linux 의 설치를 원활하게 하며 체계를 잘 구성하기 위한 필수적인 단계이다.

이 장은 Linux 의 최소체계에 대하여 간단히 언급하는것으로부터 시작한다. 실용적인 Linux 체계를 실행시키려면 어떤 하드웨어환경이 요구되는가? 물론 한장의 플로피디스크를 가지고 Linux 를 기동시킬수도 있지만 이러한 체계는 여러가지 목적에 리용할수 없다. 이 장에서는 Linux 가 콤퓨터에서 쓸모 있는 체계로 되자면 어떤 장치들이 요구되는가를 설명한다.

Linux 를 설치하기전에 우선 어떤 형태의 콤퓨터가 있어야 하는가를 알아야 한다. 이로부터 이 장에서는 아주 중요한 문제인 하드웨어호환성에 대하여 론의하게 된다. 하드웨어에 있을수 있는 거의 모든 부분품들에 대한 구동프로그람을 장치제조업체들이 재빨리 제공해 주는 Windows 세계에서조차도 결함들이 있고 하드웨어의 호환성이 보장되지 않아 Windows 조작체계를 동작시키려고 밤을 새우는 경우도 있다. Linux 에서도 마찬가지로 현재 제한되였거나 지원되지 않는 하드웨어를 리용하려고 한다면 같은 문제에 부닥치게 된다.

요구사항

Linux 를 설치하기에 앞서 무슨 형태의 콤퓨터가 필요한가를 정확하게 고찰하는것이 중요하다. Linux 는 다음과 같은 하드웨어들을 리용하는 각종 하드웨어환경에 설치될수 있다.

- ARM 처리소자
- Motorola 68000 계렬처리소자
- 8086 처리소자
- Alpha 처리소자
- SPARC 처리소자
- MIPS 체계
- PowerPC 방식체계
- S/390 방식봉사기
- Acorn 콤퓨터
- Power Macintosh
- Intel 및 Intel 호환개인용콤퓨터

최근에는 Intel 호환개인용콤퓨터가 가장 일반적인 Linux 가동환경으로 되고 있다. 이 콤퓨터는 일반적으로 Linux 에 대하여 가장 낮은 가격 대 성능비를 보장하며 대부분 Linux 도구들을 위한 초기개발가동환경이다. Intel Linux 는 주변장치들에 가장 알맞는

장치구동프로그람들과 함께 상품 및 무료형식의 아주 광범한 쓸모 있는 응용프로그람, 인터네트의 지원과 방조에 의거하는 매우 강력한 사용자집단을 제공한다.

이러한 리유로 하여 이 책은 Intel 호환의 x86 에 기초한 콤퓨터용 Linux 에 중심을 두고 있으며 이 책의 부속물인 CD-ROM 은 Intel 가동환경용 Red Hat Linux 7.1 을 포괄하고 있다.

Linux 를 위한 최소규모의 개인용콤퓨터

조작체계로서의 Linux 는 놀랄만큼 간단한 요구사항을 제기한다. 4MB 의 RAM 을 가진 386 콤퓨터에서도 Linux 를 기동시킬수 있다. 물론 이러한 콤퓨터들은 다음과 같은 제한성을 가진다.

- X Windows 를 실행시킬수 없다. 즉 GUI 가 아니다.
- 동시에 실행시킬수 있는 프로그람의 수가 물리적인 RAM 용량에 따라 제한된다.
- 속도가 느리므로 중요한 사명을 지닌 전자우편편의프로그람이나 Web 편의프로그람과 같은 응용프로그람들을 실행하기 어렵다.

이러한 제한조건이 있는 체계도 다음의 분야에서는 리용할수 있다.

- 응용프로그람이 실행되고 있는 기타 Linux 봉사기의 말단 및 Unix 봉사기의 말단
- DNS(www.yahoo.com 과 같은 주콤퓨터이름을 실지 IP 주소로 변환하는것을 방조하는 령역이름봉사)나 소규모망을 위한 인증봉사기와 같은 봉사를 위한 하부말단봉사기

사실상 Linux 는 이러한 낡은 장치환경을 가진 DOS 보다 더 우월한 기능들을 제공할수 있다. DOS 는 망기능이 제한되여 있으며 우에서 설명한것과 같은 봉사기의 역할을 수행할수 없다.

만일 이러한 최소판의 Linux 체계를 쓰고 싶다면 http://www.linuxdoc.org/HOWTO/mini/Small-Memory에서 Small Memory Mini-HOWTO 를 보고 제한된 기억기를 가진 체계에서 Linux 를 리용하는데 도움이 되는 기초자료들을 얻을수 있다.

Linux 에 좋은 PC

이 책의 기본목적은 개인용콤퓨터에서 혹은 작은 인트라네트봉사기로 Linux 를 기동시키는것이다. 불필요한 말이지만 사용자들은 앞에서 서술한 그런 형태의 콤퓨터에서 Windows 를 기동하려고 하지는 않으며 Linux 의 특성과 성능을 충분히 발휘할수 있는 더 좋은 PC 를 요구한다.

Linux 는 보통의 Windows 98 이나 Windows NT/2000 체계보다 더 많은 기능을 수행하지만 훨씬 적은 자원들을 요구한다. 실례로 기능상 워크스테이션은 486-100MHz 처리소자와 16MB 의 RAM 으로 구성될수 있다. 이 체계는 도형대면부를 위한 X Windows 를 기동할수 있고 인터네트를 호출할수 있으며 도형방식의 Web 열람기도 실행시킬수 있고 한편으로는 망에서 하부말단봉사기로서의 역할도 수행할수 있다.

일반사용자는 여전히 보다 강력한 Linux 체계를 요구한다. 괜찮다고 볼수 있는 Linux 워크스테이션은 다음과 같은 제원을 요구한다.

Pentium 급 CPU Pentium 133 도 대다수의 사용자들에게 적합하다. Cyrix 686 계렬과 같은 특정한 모방제품은 피하는것이 좋다. 일반적으로 대부분의 Pentium 급체계는 괜찮게 동작한다. 물론 오늘에 와서 새로운 PC 를 사려고 한다면 표준 Pentium 은 찾을수 없으므로 Pentium IV, Celeron, Athlon, Duron, Itanium 급체계를 선택해야 한다.

MB 의 RAM Linux 도 기억기를 확장하면 좋다. MB 는 보통 워크스테이션에 대해서 충분하지만 64MB 이상의 RAM 을 설치하면 그 차이에 대하여 알수 있다.

3GB 하드디스크 1GB(혹은 그이하도 되지만)의 하드디스크도 충분하지만 더 여유가 있는것이 좋다. 용량이 큰 디스크는 낡고 용량이 작은 원판보다 더 잘 동작할수 있다. 적어도 3GB 의 하드디스크공간이 없으면 많은 경우 Red Hat Linux 7.1 과 관련된 모든 기능들을 모두 설치할수 없다.

지원되는 비데오기판 뒤에서 보게 되는 하드웨어호환성에 대한 절을 참고하면 된다.

이와 같은 체계가 있으면 탁상조작체계인 Linux 를 기동할수 있는 아주 충분한 자원을 가지게 된다. Linux 의 성능을 높이고 빠른 속도로 실행하게 한다고 하면서 요란히 떠드는 최신판 1.7GHz 의 Pentium IV 를 구입하지 않아도 된다. 사실상 판매된지 몇주 되지 않은 최신판하드웨어는 피해야 한다. 더 자세한 정보에 대해서는 후에 나오는 하드웨어호환성절에서 설명한다.

추가적인 장치

물론 오늘의 콤퓨터환경에서는 다매체와 인터네트와 같은 령역으로 PC 능력을 확장할것을 요구한다. 일부 Linux 체계들을 크게 개선하는 몇가지 추가품들이 있으므로 워크스테이션을 보강하기 위한 방법들을 고려해야 한다. 즉

CD-ROM 구동기 이것을 설치하려면 IDE/ATAPI, 가능하다면 SCSI CD-ROM 구동기를 선택해야 한다. 자기자체의 결합기판을 가지고 작업하거나 음성기판에 특수한 결합부로 직접 련결된 CD-ROM 구동기를 피하는것이 좋다. 이러한 구동기들은 보통 성능이 낮으므로 Linux 에서 설정하기 힘들다.

음성기판 대부분의 Sound Blaster 호환기판들이 Linux 에서 지원된다. 이것도 하드웨어호환성에 대한 부분에서 볼수 있다.

모뎀 속도측면에서 보면 Windows 에서와 같은 규칙이 여기서도 적용된다. 일반적으로는 ISP(인터네트봉사소)에 빠른 속도로 접속될수 있는 가장 빠른 모뎀을 얻는것이 좋다(ISP 가 56Kbps 접속을 제공할수 없으면 56Kbps 모뎀을 구입할 필요가 없다). 두가지 조언을 준다면 우선 Linux 에서는 외장형모뎀을 선택하는것이 좋다. 특히 ISDN 모뎀을 실례로 들수 있는데 그것은 내장형 ISDN 모뎀에 대한 Linux 의 지원이 제한되기때문이다. 외장형모뎀(ISDN 혹은 상사형)의 우점은 설정이 쉽고 외부표시기를 가지기때문에 사용자가 설정오유를 쉽게 찾아 낼수 있는것이다. 둘째로는 Microsoft Windows 구동서고파일을 리용하는《winmodem》을 피하는것이 좋다. Linux 는 몇가지 winmodem 에 대해서만은 어렵지 않게 지원한다.

이처럼 CD-ROM, 음성기판, 모뎀은 최근 PC 에 대한 표준장치로 된다. 만일 Linux 를 인트라네트우의 작은 봉사기로 리용할 계획이라면 다음의 추가품들에 대해서도 고려해야 한다.

SCSI 기판 SCSI 는 IDE 결합부보다 하드구동기들에 훨씬 더 좋은 성능을 제공하며 다중장치들도 지원한다. 만일 어떤 형태의 다중사용자체계(실례로 파일봉사기, Web 봉사기, 응용프로그람봉사기)를 기동하려면 SCSI 기판이 반드시 있어야 한다. SCSI 기판을

선택하기전에 하드웨어호환성절을 참고하고 가능하다면 Ultra-Wide SCSI 를 지원하는 기판을 선택할수 있다.

SCSI 하드구동기 SCSI 기판의 한가지 기능은 속도가 빠른 SCSI 하드구동기들을 리용할수 있는것이다. 가장 좋은 성능을 보장하려면 될수록 Ultra-Wide SCSI 구동기를 리용해야 한다. 여러개의 디스크구동기를 생각할수도 있다. 실례로 조작체계와 설치된 모든 응용프로그람, 사용자의 자료를 보관할수 있는 8GB 의 공간이 필요하다면 2 개의 4GB 구동기(하나는 자료용이고 다른 하나는 조작체계 및 쏘프트웨어용)를 리용할수 있다. 쏘프트웨어와 자료를 분리하면 두 디스크를 동시에 호출할수 없으므로 성능이 높아 지는것을 볼수 있다.

테프구동기 봉사기를 기동하려면 체계의 오유나 기타사고로부터 자료안전을 보장하기 위하여 여벌복사를 진행하여야 한다. 이 경우 플로피디스크로 련결되는 테프구동기를 리용할수도 있지만 SCSI 테프구동기를 선택하면 성능 더 좋아 진다. Linux 에서 이것들은 더 빨리 동작하며 더 잘 지원된다.

하드웨어의 호환성검사

Windows(특히 Windows NT/2000)체계를 접할 때처럼 하드웨어를 구입하기 위해 자금을 투자하기전에 자기가 사려는 하드웨어가 Linux 조작체계우에서 동작할수 있으며 자기의 콤퓨터우에서 안정하게 돌아 갈수 있는가를 검사하는것이 중요하다. 하드웨어가 조작체계와 다른 하드웨어와 호환이 맞지 않으면 시끄운 일이 생길수 있으며 이로 하여 오유를 퇴치하고 콤퓨터를 재구성하는데 많은 시간을 랑비할수 있다.

이 문제는 Linux 세계에서 특별히 중요하다. Linux 를 지원하는 하드웨어제작자들이 많아 지고 있지만 아직까지도 많은 제작자들이 자기의 하드웨어를 위한 Linux 구동프로그람을 제공하지 못하고 있는데 그 제작자들가운데는 Linux 환경에서 하드웨어충돌과 기타 문제점들을 퇴치할수 있는 능력이 부족하거나 그것을 바라지 않는 사람도 있다. 이것은 하드웨어가 그 사용자의 Linux 배포물에 포함된 구동프로그람 혹은 문제시되는 하드웨어를 위한 구동프로그람을 제공하는 추가쏘프트웨어에 의해 지원되여야 한다는것을 의미한다. 게다가 사용자들은 문제들이 생길 때 Linux 세계의 방조를 받아야 한다.

하드웨어의 대부분은 Linux 에 의해서 몇가지 방법으로 지원되는데 그것은 Linux 를 설치하기전이나 Linux 체계용의 새로운 하드웨어를 구입하기전에 일련의 조사를 진행할것을 요구한다. 다음과 같은것들을 조사해 볼수 있다.

- hardware.redhat.com 에 있는 Red Hat Hardware Compatibility List 를 참고해야 한다. 여기에는 최신 Red Hat 배포물들과 호환되는 하드웨어를 탐색할수 있는 자료기지가 들어 있다.
- Linux Hardware Compatibility HOWTO 를 참고해야 한다. Patrick Reijnen 이 인증한 이 문서에는 Linux 에서 동작하는 하드웨어와 동작하지 못하는 하드웨어 그리고 이 두 형태의 하드웨어와 관련된 문제점들에 대한 확장목록이 들어 있다. 이 HOWTO 문서고에 있는 하드웨어를 구입하면 설치하기가 더 쉽다. http://www.linuxdoc.org/HOWTO/Hardware-HOWTO.html 에서 이 설명서의 최신판본을 찾을수 있다.
- comp.os.linux.hardware 새소식그룹을 참고해야 한다. 이것은 Linux 와 관련된 하드웨어에 대한 좋은 정보원천이다. 구입하려는 하드웨어가 광범하게 리용되는것인지 확신할수 없으면 그 하드웨어에 대해 알고 있는 그룹에 물어 보아야

한다. 그러면 보통 고심하던 문제를 다른 사람들이 해결하였고 공동의 지혜는 구입을 결정하는데서 대단히 중요한 자원으로 된다는것을 알수 있다. 대신에 http://groups.google.com에 있는 새소식그룹문서고들을 조사할수도 있다.

- 하드웨어를 구입하기전에 그것을 평가해 보아야 한다. 자기가 정기적으로 구입하여 리용하고 있는 하드웨어의 제작자로부터 하드웨어를 구입하려 할 때에는 실지로 그것을 사기전에 그 하드웨어를 빌려서 Linux 에서 시험해 볼수 있다. 물론 이것은 자기가 원하는 하드웨어가 틀림없이 동작할수 있다는것을 담보하는 경우이다.

하드웨어정보의 기록

목적한 LinuxPC 를 동작시키려면 자기의 하드웨어가 작업할수 있는 하드웨어관련정보를 수집하여야 한다. 이 절에서는 Linux 로 빨리 작업하도록 하는데서 알고 있어야 할 하드웨어의 정보들에 대하여 간단히 설명한다.

비데오기판

X Windows(Unix 체계의 도형사용자대면부)가 없이 Linux 를 설치한다면 임의의 비데오기판을 써도 아마 어려운 점이 없을것이다. 그러나 X Windows 에서 설치한다면 기판을 동작시키는 세부적인것들에 주의를 돌려야 한다. Linux 를 설치하기전에 비데오기판에 대한 다음과 같은 정보를 기록하여야 한다.

- 기판의 제작자와 모델
- 기판에 리용한 비데오소편(X Windows 는 어떤 특수한 기판에 대해서 명백히 지원하지 못하는 경우도 있지만 일반적으로는 기판에 있는 소편을 지원한다.).
- 기판의 비데오기억용량
- 기판에 있는 박자소편의 형태(대부분의 일반 기판들에는 박자소편들이 없다.)
- 기판에 있는 RAMDAC 형태(대부분의 일반 기판들에는 RAMDAC 들이 없다.)

이 모든 정보들은 기판에 따르는 문서 혹은 Linux 와 련관된 파일들, HOWTO 혹은 기판제작자의 Web 싸이트에서 볼수 있다.

음성기판

음성기판들이 정상동작하자면 그에 필요한 매우 특수한 정보가 있어야 한다. 대부분의 음성기판들을 구성하기 위해서는 다음의 정보가 있어야 한다.

- 기판의 제작자와 모델
- 기판의 IRQ(한개이상)
- 기판의 I/O 주소(한개이상)
- 기판의 DMA 주소(한개이상)

건늠(Jumper)스위치나 DIP 스위치를 리용하여 IRQ, I/O 주소 및 DMA 주소를 수동으로 설정할수도 있다. 이러한 기판방식에서는 이 모든 정보가 필요 없을수도 있다. 기판설치와 관련된 문서를 참고해야 한다.

현시장치

비데오기판과 마찬가지로 현시장치가 X Windows 에서 최적으로 작업하게 하자면 현시장치의 기술적특성을 기록해야 한다. 만일 이런 정보가 없거나 틀린 정보를 리용하면 현시장치가 위험하게 될수 있다.

현시장치의 문서를 참고한후에 다음과 같은 특성들을 기록해야 한다.

- 현시장치의 제작자와 모델
- 현시장치의 최대해상도
- 최대해상도에서 현시장치의 재생속도
- 현시장치의 수평동기범위
- 현시장치의 수직동기범위

X Windows 의 구성에 적용할수 있는 수평 및 수직동기범위에 대해서는 제 7 장 《X Windows 의 설치와 구성》에서 본다.

마우스

두가지 경우 즉 Linux 의 문자방식조종탁과 X Windows 에서 마우스로 작업하려면 다음과 같은 정보들에 주의를 돌려야 한다.

- 마우스의 제작자와 모델
- 마우스의 단추수
- 마우스규약(이에 대해서는 마우스문서를 참고할수 있다. 일반규약에는 Microsoft 규약, USB, Mouse System 규약 및 PS/2 규약들이 있다.)
- 콤퓨터와 접속하기 위한 마우스포구(이것은 보통 DOS 술어로 COM1:, COM2: 혹은 PS/2 마우스포구이다.)

하드구동기

설치과정에 하드구동기를 재구성할 때 Linux 를 리용하려면(이 장의 마감에 있는 하드디스크구획의 할당을 보시오.) 다음의 정보들이 있어야 한다.

- 하드디스크의 총 적재능력
- 실린더수
- 자두의 수
- 자리길당 분구수

일반적으로 Linux 체계가 기동할 때 하드디스크를 자동검출하므로 이러한 정보를 제공할 필요는 없다.

모뎀

모뎀을 가지고 있다면 다음의 정보를 기록해야 한다.

- 모뎀의 제작자와 모델
- 모뎀의 속도
- 외장형모뎀이 련결되였거나 내장형모뎀을 구성하여 리용하는 포구(이것은 보통 DOS 술어로 COM1: 혹은 COM2:이다.) 혹은 포구뿐아니라 내장형모뎀들을 가진 IRQ, I/O 주소

망기판

망기판을 가지고 있다면 다음의 정보를 기록해야 한다.

- 망기판의 제작자와 모델
- 기판의 IRQ(한개이상의)
- 기판의 I/O 주소(한개이상의)
- 리용가능한 특수화된 구동프로그람
- Novell 1000 혹은 2000 망기판구동프로그람들과의 호환성

일부 망기판들은 Linux 구동프로그람들이 있는 구동디스크들을 갖추고 있다. 한편 자기의 기판이 Novell 1000 혹은 2000 망기판구동프로그람들과 호환할수 있으면 Linux에 망기판을 설치할 때 이 구동프로그람들을 리용할수 있다. 더 자세한 정보에 대해서는 망기판문서를 참고할수 있다.

USB

일부 Linux 배포물들은 건반과 마우스를 대상으로 하는 USB 지원들을 가지고 있다. 이러한 배포물들에는 S.u.S.E 6.4, Red Hat 7.0, Mandrake 7.1, Corel Linux 2 판(그 이상)이 포함된다. 2.2.x 핵심부에서 USB 하드웨어를 위한 Linux 지원은 정적구성으로 제한된다.

그러나 Red Hat 7.1은 2.4.x 핵심부에 기초하고 있으며 이 핵심부는 USB 지원을 위한 충분한 사용공간을 제공한다. http://www.qbik.ch/usb/devices 의 Linux-USB 장치자료기지에서 보여 준것처럼 모뎀들로부터 Web 카메라에 이르는 모든 형태의 USB 장치준위를 리용할수 있는 Linux USB 지원기능도 있다.

만일 콤퓨터조작시 USB 장치를 설치할수 있는 《hotplugging》에 흥미를 가진다면 http://linux-hotplag.sourceforge.net에 있는 Linux Hotplugging Web 싸이트를 참고할수 있다. 이 싸이트에는 Red Hat Linux로 작업하도록 설계되고 rpm 형태로 내리적재할수 있는 스크립트들이 있다.

Linux 에서 USB 를 리용하기 위한 연구가 진행되고 있다. 만일 자기의 배포물에 USB 하드웨어를 설치할수 없다면 추가적인 구동프로그람들과 편의프로그람들을 볼수 있는 http://www.linux-usb.org에서 Linux USB 대상과제를 다시 조사해야 한다.

기타 주변장치

많은 주변장치들을 구성할 때 특별한 요구들이 제기된다. 그것들은 너무도 많아서 가능한 주변장치들의 수와 형태를 이 책에서 다 설명할수는 없다. 이러한 주변장치들에 대한 구성은 하드웨어의 형을 론의하는 해당한 절들에서 더 구체적으로 설명한다. 일반적으로 전문화된 직렬기판들, 전문화된 PCMCIA 혹은 PC Card 들과 테프구동기와 같은 추가적인 하드웨어는 Linux 가 설치될 때 구성 및 설치되는것이 아니라 Linux 체계가 실행하는 상태에서 설치된다.

만일 국부망에서 Linux 체계를 설치하려면 제 27 장《SOHO를 위한 Red Hat Linux 7.1의 설치》를 참고할수 있다. 그러면 망기판들과 Linux 설치에 속하는 정보를 알수 있다.

설치방법의 선택

일반적으로 Linux는 절대적인 크기로 해서 CD-ROM으로 배포된다. Linux를 인터네트에서 내리적재할수도 있지만 그것은 너무 용량이 커서 고속인터네트접속을 리용하지 못하는 사람들은 내리적재할수 없다.

이 책에서 설명하는 Red Hat Linux 7.1배포물인 경우 CD-ROM에는 Linux를 설치하는 기본핵만이 있다. 인터네트와 떨어져서 직접 Linux를 설치하는것이 리론적으로 가능하지만 이 방법은 대다수의 Linux 사용자들에게 있어서 실천적으로 시간소비가 많고 값이 비싸다.

이 절에서는 이 책과 함께 제공된 CD-ROM으로부터 Linux를 설치하는 각이한 수법들을 고찰한다. 설치순서는 CD-ROM으로 설치할수 있는 대부분의 다른 Linux배포물들과 비슷하다. 그러므로 그러한 배포물들에 대한 문서를 보고 차이점을 결정할수 있다.

CD-ROM에 의한 설치

만일 IDE/ATAPI CD-ROM 구동기와 최신판 BIOS를 가진 콤퓨터가 있으면 그 콤퓨터를 기동하여 Linux CD-ROM으로 설치작업을 시작할수 있다.

이것을 검사하려면 콤퓨터나 주기판의 안내서를 참고하든가 혹은 콤퓨터가 기동하는 과정에 BIOS 설정을 입력하고 기정기동장치를 CD-ROM 구동기로 절환할수 있는가 본다. 만일 CD-ROM구동기로부터 기동할수 있다면 Red Hat Linux 7.1설치 CD-ROM을 넣고 체계를 기동시킨다. 그러면 설치차림표와 《boot:》라고 하는 기동재촉문이 나타난다.

주의 설치 CD가 CD-ROM으로부터 기동할수 있게 된것이라고 해도 기동가능한 CD-ROM들을 지원하는 모든 PC들에서 기동할수 있는것은 아니다. 만일 이 책과 함께 제공된 Linux의 복사물이나 자체로 복사한 CD로 구동하기 힘들면 다음절 《플로피디스크와 CD-ROM에 의한 설치》에서 언급한것처럼 플로피디스크와 CD-ROM으로 설치해 볼수 있다.

플로피디스크와 CD-ROM에 의한 설치

CD-ROM 구동기가 있지만 이것으로 기동시킬수 없다면 다음에 시도해 볼수 있는 가장 좋은 방법은 바로 플로피디스크와 CD-ROM을 결합해서 Linux를 설치하는것이다. 여기서는 먼저 한개이상의 플로피디스크로 기동하여 설치작업을 시작하고 다음에 CD-ROM으로 완전한 Linux쏘프트웨어를 설치한다.

어떤 경우에는 반드시 이렇게 해야 한다. 대부분의 Linux 배포물에서는 기동플로피디스크와 몇개의 보조디스크를 준비하는것이 좋다. 작업에 필요한 모든것은 Linux 배포물이 들어 있는 CD-ROM에 있다.

Linux 설치 CD-ROM들에는 관계없이 기동디스크와 기타보조디스크를 요구한다면 DOS나 Windows DOS재촉문에 그 디스크들을 입력할수 있다.

Red Hat 7.1 CD-ROM의 Images 보조등록부에는 두개 파일 즉 boot.img와 bootnet.img가 있는데 이것들은 국부콤퓨터의 파일이나 망을 통하여 Red Hat Linux 7.1를 설치하는데 리용되는 플로피디스크의 디스크영상(disk images)이다. 거기에는 또한 여러가지 PCMCIA 카드용구동프로그람디스크 (pcmcia.img와 pcmciadd.img), 낡은

CD-ROM 용구동프로그람(oldcdrom.img) 및 기타구동프로그람(drivers.img)이 있다. 이 매개 영상은 초기화된 1.44MB 플로피디스크와 rawrite.exe 편의프로그람을 요구한다.

실례로 CD-ROM 구동기가 D 구동기라면 Red Hat 기동설치디스크를 만들기 위해 rawrite.exe 를 리용한다. 즉

C:\ >**d:\ dosutils\ rawrite.exe**
Enter disk image source file name: **d:\ imges\ boot.img**
Enter target diskette drive: **a**
Please insert a formatted diskette into drive A: and press **-ENTER-** :

마찬가지로 망기동설치디스크를 만들려면 원천파일이름으로 **d:\ images \ bootnet.img** 를 입력해야 한다.

이 과정이 끝나면 기동설치디스크를 기동해서 설치작업을 시작할수 있다.

하드디스크에 의한 설치

충분한 디스크공간이 있다면 자기의 하드디스크에 CD-ROM 의 전체 내용을 복사하고 하드디스크로 설치할수 있다. 하드디스크로 설치할 때 Red Hat Linux 7.1 이 기동하면 Red Hat 설치 CD-ROM 의 모든 파일들을 포함하는 단일 파일인 ISO 영상을 리용해야 한다.

정확한 ISO 영상은 두가지 방법으로 얻을수 있다. 즉 그것을 인터네트에서 직접 내리적재할수도 있고 ftp.redhat.com 과 같은 원천파일에서 내리적재할수도 있다. Red Hat Linux 7.1의 매개 CD-ROM에 있는 ISO파일은 CD-ROM만큼 크다. 다시말하여 대략 650MB 이다. 한편 mkisoft 지령을 리용하여 Red Hat Linux 7.1 의 매개 CD-ROM 으로부터 한개의 ISO 영상을 만들수 있다. 이 과정에 대한 상세한 정보는 제 5 장 《특수설치》에서 볼수 있다.

물론 ISO영상을 만들기 위해서 한개의 CD-ROM구동기를 호출한다면 하드디스크로 설치할 필요가 없다. 이것이 실제로 필요한 때는 Linux설치쏘프트웨어가 CD-ROM구동기를 인식하지 못하는 경우이다. 그리고 이미 언급된 oldcdrom.img 파일을 리용할수 있는 구동프로그람들이 있으면 이것은 드문히 요구될수 있다.

하드디스크의 배렬

Linux 를 설치할 때가 되면 하드디스크의 어디에 조작체계를 설치하겠는가에 대해서 기본적으로 결심을 내려야 한다. 기회가 아주 좋은 두가지 경우가 있을수 있다.

- Linux 를 설치할수 있는 빈 하드디스크나 다시 양식화할수 있는 하드디스크가 있다.
- Linux를 설치할수 있는 빈 구획이나 다시 양식화할수 있는 구획들이 있다.

처음으로 Linux 를 설치하는 대다수의 사용자들은 Linux 가 현재 자기들의 Windows 나 DOS 설치들과 공존하기를 바라며 현존구획이나 하드디스크를 다시 초기화

하는것은 바라지 않는다.

구획짓기의 개념

빈 구획이나 하드디스크가 없는 현행체계에 Linux를 설치하려면 충분한 디스크공간을 찾고 Linux 설치과정에 리용할 구획을 개방하도록 체계의 구획짓기를 세밀하게 조종해야 한다. 보통 완전한 Linux체계를 설치하려면 적어도 2GB의 디스크공간을 개방하여야 한다. 개방한 공간은 한 구획에 있어야 한다(Windows에서는 매개 구획이 C, D, E와 같이 개별적인 구동기문자로 표시되므로 2GB이상의 빈 공간을 가진 구동기를 찾아야 한다).

Windows 98체계를 위한 표본구획방식

간단한 실례를 보자. DOS에서 두개의 6GB구획 즉 구동기 C와 D로 분리된 12GB의 하드디스크를 갖춘 콤퓨터가 있다. 구동기 D에서 3GB의 공간을 개방하여 Linux를 설치하는데 리용할수 있다.

Linux를 설치하기전에 두 단계를 거쳐야 한다.

1. 구획의 마지막에 크고 련속적인 빈 공간령역이 놓이도록 하기 위하여 구동기를 재정리(Defragment)한다.
2. Linux설치에 필요한 공간을 만들기 위하여 구동기를 재분할한다.

구동기의 재정리

Windows 98에서 구동기를 재정리하기는 아주 간단하다. 즉 다음과 같은 단계를 거친다.

1. 구동기에 있는 자료들을 여벌복사한다.
2. My Computer나 Windows Explore에서 Drives아이콘을 찰칵한다.
3. 내림차림표에서 Properties를 선택한다.
4. Properties창문에 있는 Tools표쪽을 선택한다.
5. Defragment Now 단추를 찰칵한다. 재정리가 끝나고 다음항목으로 넘어 갈 때까지 기다린다.

디스크의 구획짓기

Linux를 설치하는데 충분한 공간을 가질만큼 구동기가 재정리되였다면 그 빈 공간에 새로운 구획을 만들어야 한다. Red Hat Linux 7.1을 비롯한 대부분의 Linux배포물들은 dosutils 보조등록부나 이 등록부의 보조등록부에 있는 fips.exe 이라고 하는 DOS 용도구도 갖추고 있다. 이 책과 함께 제공된 Red Hat 판본에서는 \ dosutils \ fips20 보조등록부의 프로그람을 볼수 있다.

경고 Red Hat는 **fips**를 지원하지 않는다. 그것은 일반공개사용허가에서 설명되는 **fips**를 리용한 사람들이 **fips**를 정확히 리용했는데도 자료가 파괴되는데 대

해 책임을 지지 않는다. 그런데 **fips**를 여러번 써보았어도 결코 조심해야 할 문제는 생기지 않았다. 즉 위험한 상태에서 그것을 리용한다. System Commander(www.v-com.com) 와 Patition Magic(www.powerquest.com)가 들어 있는 대리적인 업무용구획분할프로그람도 있다.

이 도구는 구획의 마지막부분에 있는 빈 공간을 삭제하여 구획을 더 작게 만들어서 이미 있는 구획의 크기를 조절한다. 결과 빈 공간은 또 다른 구획으로 변환된다.

fips.exe 를 리용하려면 처음에 MS-DOS 방식으로 되여야 한다. 이를 위해서 Start 차림표로부터 Shut Down 을 선택한 다음 Restart In MS-DOS Mode 를 선택한다. Windows 98 은 닫기며 완전한 화면방식(full-screen)DOS 환경으로 절환된다.

경고 이것은 위험한 단계이다. Windows 98 이 실행하는 DOS 창문이나 완전한 화면 방식 DOS 환경에서 **fips.exe** 와 같은 프로그람은 리용하지 말아야 한다. DOS 가 아닌 Windows 95 와 그이상의 체계들에서는 여러가지 프로그람이 동시에 실행될수 있으므로 **fips.exe** 가 작업하고 있는 구획을 호출할수 있다. 그러나 그렇게 되면 자료들이 못 쓰게 되거나 살려 낼수 없다.

일단 DOS 방식으로 넘어 가면 CD-ROM 으로 fips.exe 를 실행해야 한다. 만일 CD-ROM 구동기가 E 구동기이면 다음과 같은 지령을 줄수 있다.

```
c:\>e:\dosutils\fips20\fips.exe
```

그 프로그람은 우선 구획표를 현시한다. 두 구획구동기들에 대한 앞의 실례에서 이 프로그람을 실행하면 다음과 같은 구획이 나타난다.

구획표

Part.	Boot able	Start Head Cyl. Sector	System	End Head Cyl. Sector	Start Sectors	Number of Sector	MB
1	yes	1 0 1	06h	254 222 63	63	35824	1749
2	no	0 223 1	05h	254 286 63	3582495	1028160	502

대부분의 체계에서 C 구동기가 구획 1, D 구동기가 구획 2 이므로 구획을 식별하기가 쉽다. 혼동되지 않게 하기 위해서 마지막항목에 MB 단위로 표시한 구획크기를 지정한다.

다음구획을 선택하라는 지령재촉문이 나타난다. 이 경우에 D 구동기를 조정하려면 2 를 선택해야 한다. 일단 선택되면 그 구획이 주사되며 구획에 대한 총적인 정보가 다음과 같이 현시된다.

Bytes per sector: 512
Sector per cluster: 8

Reserved sectors: 1
Number of FATs: 2
Number of rootdirectory entries: 512
Number of sectors(short): 0
Media descriptor byte: f8h
Sectors per FAT: 145
Sectors per track: 63
Drive heads: 16
Hidden Sectors: 63
Number of sectors(long): 141057
Physical drive number: 80h
Signature: 29h

선택한 구획의 마지막부분에 빈 공간이 있다고 보면 구획을 가르고 분할경계로 어느 디스크원기둥을 리용하겠는가 하는 질문을 나타난다. 선택된 원기둥을 변경시키기 위해서 방향건을 리용할수 있다. 결과 MB 단위의 구획들이 나타나며 따라서 새 구획이 충분히 크다는것을 정확히 확인할수 있다. fips.exe 프로그람은 새로운 구획에서 현행구획의 일부 자료들이 남아 있는 쪼각으로 원기둥을 선택할수 없겠는지를 확인하는 일감을 조종한다.

마지막으로 수정된 구획표가 현시되며 모든것이 정상인가를 확인하라는 질문이 나타난다. 이 실례에는 현재 3 개의 구획들이 있는데 세번째는 새 구획이고 두번째는 처음보다 크기가 더 작다.

일러두기 오유를 방지하려면 Linux 설치 CD-ROM 에 있는 **fips.exe**, **restorrb.exe** 와 **errors.txt** 파일을 기동할수 있는 플로피디스크에 복사해야 한다. 이 파일들은 보통 **dosutils** 와 같은 보조등록부에 있다. 만일 플로피디스크에 boot 와 root 분구들에 대한 여벌복사물들을 써넣겠는가고 물으면 **Yes** 라고 대답한다. 그다음 오유를 다시 찾을 필요가 있으면 플로피디스크로 기동하고 초기시동항목과 뿌리항목을 다시 보관하기 위하여 **restorrb.exe** 를 실행한다.

▶미리 알아두기

이 장에서는 Linux 를 설치하는데 필요한 기본준비를 고찰하였다. 이제는 실제로 설치할 준비가 되였다고 할수 있다.

제 4 장 《Red Hat Linux7.1 의 설치》에서는 설치프로그람의 매개 화면과 그것을 만드는데 무엇을 선택해야 하는가를 결정하는 방법을 고찰하고 Linux 의 기본설치를 배우게 된다.

제 5 장《특수설치》에서는 여러개의 하드구동기로 Linux 를 설치하는 방법과 이미 있는 DOS 구획에 Linux 를 설치하는 방법과 같은 특수한 경우를 고찰한다.

제 4 장. Red Hat Linux 7.1 의 설치

이제는 Linux 를 설치할수 있다. 지금까지의 단계를 통하여 Linux 설치 **CD-ROM** 을 기동하거나 혹은 Linux 기동플로피디스크들을 만들수 있어야 한다. 또한 Linux 를 설치하기 위한 장소를 결정하고 필요한 공간을 조성할수 있어야 한다.

이제는 시작할수 있다. Linux 와 같은 조작체계를 써본적이 없다면 설치프로그람이 제기하는 질문들이 낯설수 있지만 사실상 모든 처리는 매우 단순하며 설치쏘프트웨어의 많은 질문들에 대해 정확한 결심들을 내릴수 있다.

사실 Windows 95/98/ME 체계에 두번째 조작체계로 Linux 를 설치하려는 일반 사용자들에게 있어서 프로쎄스는 일반적으로 약간의 방향을 돌리면 되는 정도로 아주 간단하다. 이 장은 Red Hat Linux 7.1 설치에 기초하며 그 원리들은 거의 모든 Linux 배포물들의 최신판본의 원리와 같다.

설치시작

이 장에서는 간단한 설치 즉 단독으로 있는 PC 에서 CD-ROM 으로 Linux 를 설치하는 방법을 고찰한다.

이를 위해서는 설치를 시작할 때부터 기동에 필요한 매체를 선택해야 한다. 앞의 장에서 언급한것처럼 두가지 방도가 있을수 있다. 즉

- Linux 설치 CD-ROM 로 직접 기동한다. 이것은 콤퓨터의 BIOS 가 CD-ROM 구동기로 기동하는것을 지원할것을 요구한다.
- 먼저 기동플로피디스크들로 기동하고 다음에 CD-ROM 으로 설치한다.

어느 경우에나 기동디스크(플로피디스크 혹은 CD-ROM)를 넣고 콤퓨터를 기동해야 한다. 콤퓨터가 Red Hat Linux 7.1 플로피디스크 혹은 CD-ROM 으로 기동하기 시작하면 처음에 그림 4-1 과 같은 안내화면이 나타난다.

```
                Welcome to Red Hat Linux 7.1!

 - To install or upgrade Red Hat Linux in graphical mode,
   press the <ENTER> key.

 - To install or upgrade Red Hat Linux in text mode, type: text <ENTER>.

 - To disable low resolution mode, type:  lowres <ENTER>
   Press <F2> for more information about low resoution mode.

 - To disable framebuffer mode, type:  nofb <ENTER>.
   Press <F2> for more information about disabling framebuffer mode.

 - To enable expert mode, type: expert <ENTER>.
   Press <F3> for more information about expert mode.

 - To enable rescue mode, type:
   Press <F5> for more information about rescue mode.

 - If you have a driver disk, type:  linux  dd  <ENTER>.

 - Use the function keys listed bellow for more information.

[F1-Main] [F2-General] [F3-Expert] [F4-Kernel] [F5-Rescue]
boot:
```

그림 4-1. Red Hat Linux 7.1 설치기동화면

여기에는 7개의 기본선택항목들이 있다.

- 표준적인 도형방식을 설치하고 처리하기 시작하려면 Enter건을 눌러야 한다.
- 이 항목이 차림표에서 명백히 보이지 않으면 설치쏘프트웨어가 기동시에 일부 형태가 불명확한 하드웨어를 검출하게 하는 파라메터들을 입력한 다음 표준설치 처리를 시작하기 위하여 Enter건을 누른다(대부분의 장치들은 설치과정에 자동적으로 검출된다.).
- 본문방식으로 기동하려면 text를 입력하고 Enter건을 눌러야 한다(본문방식에서는 도형방식화면이 없는 상태에서 설치를 시작한다. Linux로 영상기판(video card)을 검출하기 힘들면 본문방식으로 넘어 가는것이 좋다. Red Hat Linux 6.0과 그이하 판본들에서 본문방식화면들을 인식할수 있다.).
- 저해상도방식으로 기동하려면 lowers를 입력하고 Enter를 눌러야 한다. 이 방식에서는 표준설치 800 * 600화면이 아니라 640 * 400화면에서 도형방식을 설치하기 시작한다. 영상기판에 800*600도형방식을 지원할만한 충분한 기억기가 없으면 저해상도방식을 선택할수 있다.

주의 방금 서술한 본문방식에서의 설치와 비교해 보면 다음에 보게 되는 두가지 설치방식이 표준도형방식설치처리에 더 가깝다.

- 비프레임완충방식으로 기동하려면 nofb를 입력하고 Enter건을 눌러야 한다. 이 방식에서는 가속화된 도형처리기판과 결합된 프레임완충기가 없이 도형방식을 설치하기 시작한다. 도형기판이 2-D나 3-D의 가속화된 도형처리를 지원하지 못하면 비프레임완충방식을 선택할수 있다.
- 전문가방식으로 기동하려면 expert를 입력하고 Enter를 눌러야 한다(전문가방식에서는 하드웨어검출이 진행되지 않으며 설치과정에 모든 장치들에 대한 구성파라메터들을 줄 필요가 없다. 장치들이 완전히 검출되지 않는다면 이 방식으로 전환하여 입력할 파라메터들을 확인하는것이 좋다.).
- 일반적인 Linux구출디스크(rescue disk)를 기동하려면 linux rescue를 입력하고 Enter를 눌러야 한다. 이렇게 하면 일부 체계실패로부터의 회복을 방조할수 있다. 사용자는 손상되였거나 잃어 버린 구성파일들을 다시 구성할수 있는 기본적인 편집기와 도구를 가진 뿌리사용자로서 콤퓨터를 호출할수 있다. 일반적으로 없어서는 안될 구성파일을 가진 전용구출플로피디스크를 만들어야 한다. 만일 그 디스크를 잃어 버리면 구출방식을 선택할수 있다.
- Linux dd선택항목은 단독적인 항목이 아니라 표준적인 도형방식설치처리를 시작하기 위한 대리방식항목이다. Linux dd를 입력하면 설치프로그람은 특수한 장치를 위한 구동프로그람디스크를 넣을것을 요구한다. 전용구동프로그람디스크는 설치 CD-ROM의 images등록부에 있는 drivers.img파일로부터 만들수 있다.

주의 일반적으로 첫번째 선택항목을 선정하면 모든 하드웨어들이 성과적으로 검출되므로 대다수의 사용자들은 기동처리를 시작하기 위하여 간단히 Enter를 누른다.

60 초내에 건입력을 하지 않으면 마치 Enter 만 누른것처럼 자동적으로 설치를 시작한다. 이 시계를 무시하려면 방조건들(F1~F5)가운데서 임의의것을 눌러야 한다. 그러면 설치프로그람을 기동하겠는가에 대해서 결심을 내릴수 있는 시간을 얻게 된다.

설치를 위한 체계구성

설치처리는 차례로 신중히 고찰하여야 하는 몇개의 단계로 이루어 진다. 그러나 그것을 보기전에 우선 설치프로그람으로 작업할수 있는 건반사용법을 배워야 한다.

건반조종

그림 4-2 에서 보는것처럼 일반적인 설치화면에는 몇가지 요소들이 있다. 이 요소들에는 본문입구마당들과 검색칸들 그리고 조종단추들이 있다.

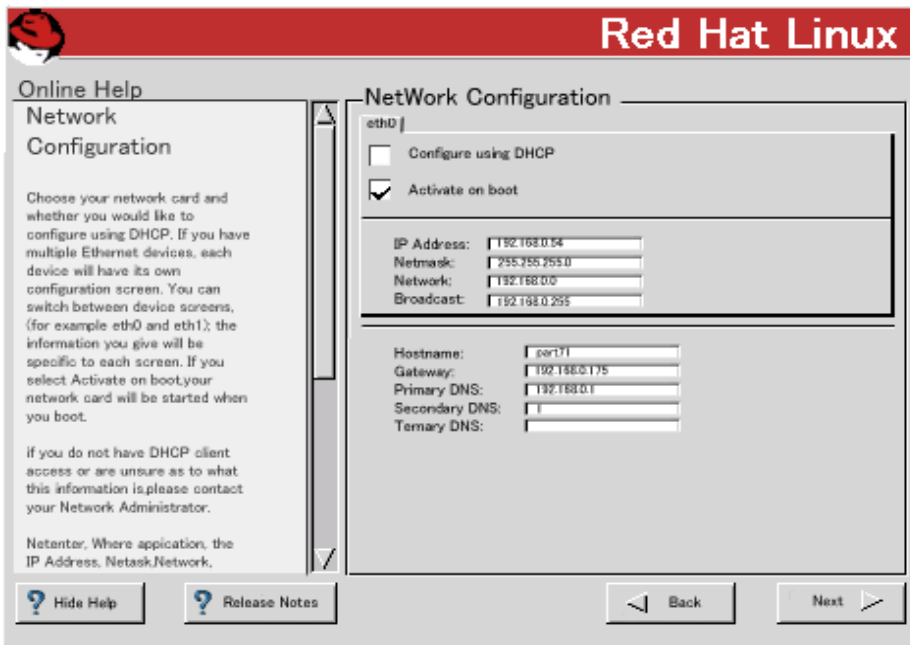



그림 4-2. 대표적인 Red Hat Linux 7.1 설치화면

이 화면에는 다음과 같은 요소들이 있다.

- 본문입구마당(IP Address, Netmask 등)
- 검색칸(Configure Using DHCP 와 Activate On Boot)
- 단추(Hide Help, Back, Next)

이러한 요소들사이의 이동은 Tab 건(혹은 반대방향으로 이동하려면 shift+Tab 건과

배합) 혹은 마우스로 할수 있다. 마당사이를 이동할 때 화살표식건들을 리용할수 있다.
검색칸의 상태를 지적하려면 검색칸에로 이동하여 공백건을 눌러야 한다. 단추를 누르려면 Tab 건 혹은 방향건들을 리용하여 단추에로 유표를 옮기고 다음에 공백건 혹은 Enter 건을 누른다. 일반적으로 Next 단추에서 F12 건은 Enter 혹은 공백건을 누른것처럼 동작한다. 물론 설치프로그람이 마우스를 검출하면 마우스의 왼쪽을 찰칵해서 선택할수 있다. 만일의 경우를 고려하여 이 장의 나머지부분에서는 마우스가 Linux 설치프로그람에서 동작하지 않는다고 보고 설명한다.

언어선택

앞에서 본것처럼 기동화면에서 설치프로그람이 기동하면 환영화면이 나타나고 뒤이어 설치처리의 첫 화면 즉 언어선택(Language Selection)화면이 나타난다. 이것은 설치처리과정에 자기가 보는 언어만이 작용하게 한다. 설치가 끝난후 Red Hat Linux 7.1 에서 언어를 변화시키려면 이 장의 마지막항목 《언어지원선택》을 보아야 한다.
그림 4-3 에서 보는바와 같이 언어선택화면에는 선택목록마당과 Next 단추가 있다. 유표가 선택목록마당에 있으면 원하는 언어를 하나 선택할 때까지 리용할수 있는 언어들로 이동하기 위하여 방향건 혹은 PageUp 와 PageDown 건을 리용할수 있다. 이 선택으로 해서 설치처리과정에 리용할 언어를 결정할수 있다(선택에 따라 제일 마지막에 선택되는 실제적인 조작체계언어와 건반배치를 다르게 할수 있다.).

그림 4-3. 언어선택

건반형식선택

다음의 화면은 건반모형과 배렬을 선택할수 있는 선택목록마당을 보여 준다. 건반이 모델목록에 없으면 그것은 기정건반으로서 Generic 104-key PC 이다. 일반적으로 건반배렬은 여러가지 언어나 방언에 따른다(실례로 프랑스 혹은 스위스). 미국에서 표준배렬은 거의나 미국식영어배렬이다. 표 4-1 과 4-2 에 준것은 다른 일반모델과 배렬이다.

표 4-1. 대표적인 건반모델

모 형	설 명
Dell 101-Key PC	Dell PC 건반
Japanese 106-Key	일본어문자를 가진 건반
Microsoft Natural	Microsoft Natural 건반
Winbook Model XP5	Winbook 특수모형 XP5 건반

표 4-2. 대표적인 건반배렬

배 렬	설 명
Brazilian	브라질식뽀르뚜갈어건반
Canadian	카나다식영어건반
German	도이취어건반
Swiss German	스위스식도이취어건반

화면아래부분의 Test Your Selection Here 본문칸에서 선택한것을 검사할수 있다. 선택한 건반에 대한 의문이 생기면 따로 검사해야 한다.

주의 가지고 있는 건반이 특수하여 Red Hat Linux 7. 1 설치프로그람의 표준방식에서 그것을 보지 못하면 설치프로그람을 본문방식으로 다시 기동해야 한다. Red Hat Linux 7.1 은 이 방식에 적어도 80 여개의 서로 다른 건반형태들을 포함하고 있다.

마우스구성

건반을 구성한 다음에는 마우스를 구성해야 한다. 설치쏘프트웨어는 우선 마우스를 검출한다. 마우스목록에서 어느것을 선택하겠는가 문의하는 재촉문이 나타난다(자동검출이 성과적으로 되면 알맞는 마우스형을 먼저 선택한다.). 목록에 요구하는 마우스형이 없으면 제일 비슷한 일반마우스를 선택할수 있다. 가지고 있는 마우스의 형을 결정하기 위해서 마우스문서를 참고해야 한다.

가지고 있는것이 두단추마우스이면 세단추마우스모방칸을 검사해야 한다. Linux 는 모든 Unix 조작체계에서와 같이 세단추마우스를 요구한다. 마우스의 왼쪽과 오른쪽단추를 동시에 찰칵하여 가운데단추의 찰칵하기를 대신할수 있다. 절대적으로 필요하지는 않지만 이러한 특성은 X Windows 의 기능을 높여 주게 된다.

주의 가운데굴개가 있는 마우스를 가지고 있으면 그것을 눌러야 한다. 만일 거기를 찰칵하면 Linux 는 그것을 세단추마우스로 인식하게 된다. 이런 경우에는 세단추마우스모방을 선택할 필요가 없다.

또한 대표적인 PS/2 형(접지를 가진 접속구)이나 직렬형(9 개의 구멍을 가진 제형접속구) 혹은 USB 형(작은 직 4 각형접속구)으로 마우스포구를 설정해야 한다. 직렬형마우스가 있으면 대체로 COM1 또는 COM2 마우스포구를 설정하거나 구성해야 한다.

설치 혹은 갱신이란 무엇인가

일단 마우스를 선택하여 구성했으면 차림표에 제시된 4 가지 방식으로 Red Hat Linux 를 설치할수 있다. 콤퓨터에 이미 Linux 가 설치되여 있으면 Upgrade 항목을 선택할수 있다. 구성파일들을 덧쓰기하지 않고 갱신하려면 보통 같은 상표가 붙은 배포물로 갱신해야 한다. 대부분의 Linux 배포물들은 같은 선택항목들을 허용하지만 설치화면에 대한 구성는 각이할수 있다.

콤퓨터에 처음으로 Red Hat Linux 7.1 을 설치한다면 4 가지 방법으로 설치할수 있다. 즉

Workstation Linux 워크스테이션에 대한 기정설치를 진행한다. 이 항목은 체계에 존재하는 모든 Linux 의 구획들을 자동적으로 삭제하며 하드구동기가 최소한 1.2GB 의 빈 공간을 가질것을 요구한다.

Server Linux 봉사기를 위한 기정설치를 진행한다. 이 항목은 Linux 가 아닌 다른 조작체계를 비롯한 체계에 존재하는 모든 구획들과 자료들을 자동적으로 지운다. 이것은 하드구동기가 최소한 650GB 의 빈 공간을 가질것을 요구한다.

Laptop 워크스테이션설치와 비슷하게 이 선택은 무릎형콤퓨터에서 Linux 를 위한 기정설치를 진행한다. 여기에는 대표적으로 PCMCIA 구동프로그람들과 같이 무릎형콤퓨터가 요구하는 프로그람묶음이 들어 있다. 이 항목은 체계에 존재하는 모든 Linux 구획들을 자동적으로 삭제한다. 이것은 하드구동기가 최소한 1.2GB 의 빈 공간을 가질것을 요구한다.

Custom 설치처리의 모든 양상들을 완전히 조종한다. 기정전용설치는 하드구동기가 700MB 이상의 충분한 빈 공간을 가질것을 요구한다.

Red Hat Linux 7.1 의 설치에 대한 더 상세한것들은 이 장의 전 과정과 제 5 장 《특수설치》, 제 27 장 《SOHO 를 위한 Red Hat Linux 7.1 의 설치》, 제 31 장 《보안과 값눅은 경로기로서의 Red Hat Linux 7.1》에서 4 번째 항목인 Custom 을 참고할수 있다.

디스크구획의 설정

설치등급을 선택한 다음 Linux 설치에 필요한 디스크공간을 어떻게 할당하겠는가를 결정해야 한다. 이것은 프로쎄스에서 어려운 단계이다. 자칫 잘못하면 보관하려고 하는 체계에 존재하는 자료가 지워 질수 있다.

제 3 장 《Red Hat Linux 7.1 설치준비》에서 본것처럼 체계에 한개의 하드디스크가 있고 거기에서 설명된 수법들을 리용하여 빈 공간의 령역(1GB 이상)을 만든 다음 그 공간을 리용할수 있는 준비가 되였다고 하자.

그러면 처음에 구획을 설정하는데 어느 도구를 리용하겠는가 하는 질문이 나타난다. Red Hat Linux 에는 Disk Druid 와 fdisk 라는 두개의 도구가 있다.

fdisk 는 디스크구획설정을 위한 Linux 의 표준도구이며 모든 Linux 배포물들에서 리용할수 있다. 그러나 리용하기 힘들고 특히 처음으로 Linux 와 Unix 를 리용하는 사용자들이 위압될수 있다.

경고	무엇을 하고 있는지 모르면 **fdisk** 는 사용자가 모든 자료를 잃어 버리게 할수 있다. 뿐만아니라 Linux **fdisk** 도구는 MS-DOS **fdisk** 도구와는 완전히 다르며 제 5 장에서 보게 되는것처럼 많은 항목들을 포괄하고 있다.

이 과정이 쉽게 진행되게 하기 위하여 Red Hat Linux 7.1 은 디스크구획을 관리하는 Disk Druid 도구를 제공한다. 이 절에서는 Disk Druid 가 Linux 환경에 새로 들어 선 사용자들에게 초기설치처리를 쉽게 해주기때문에 그것의 리용에 중점을 두고 고찰한다. Disk Druid 단추를 선택하면 그림 4-4 에서 보는것처럼 Disk Druid 기본화면이 나타난다.

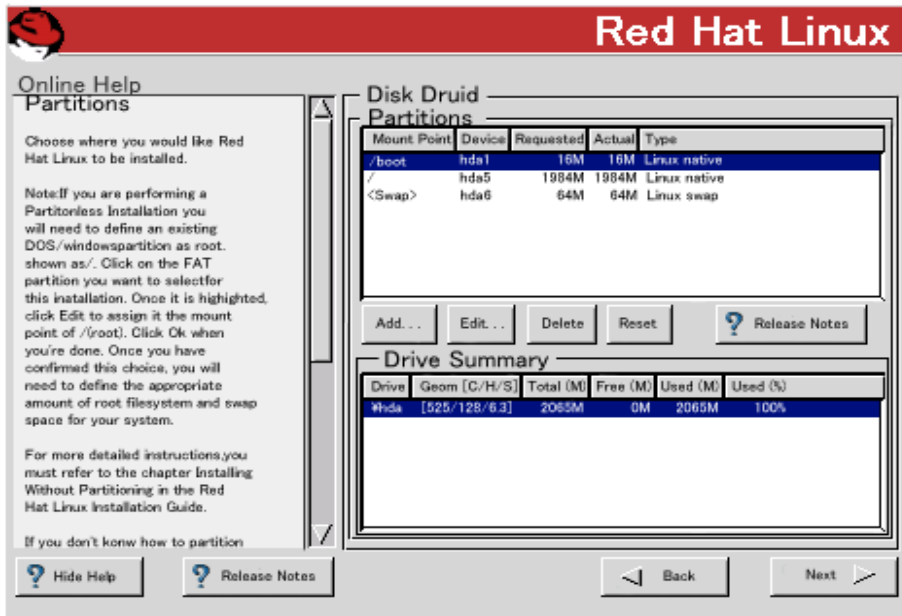



그림 4-4. Disk Druid 기본화면

이 화면은 3 개의 주요부분 즉 Partitoins 부분과 단추렬 그리고 Drive Summary 부분으로 이루어 져 있다.

Partitions 부분에는 체계에 존재하는 모든 디스크구획들이 행단위로 현시된다. 다음의 정보들은 매 구획에 제시된다.

Mount Point 이것은 구획이 Linux 등록부구조의 어디에 나타나는가를 가리킨다. Linux 등록부들은 모두 뿌리등록부의 보조등록부들로 나타난다. 뿌리등록부는 /이며 모든 보조등록부들은 /home, /opt, /usr/X11R6 과 같이 /로 시작한다. 최소한 /로 설치된

구획과 교환구획을 가져야 한다. 추가구획들이 보조등록부들에 설치되게 선택하면 이 보조등록부는 추가등록부들을 호출하도록 Linux 배포물을 전개한다. 실례로 구획을 /usr 와 같이 설정하면 이 보조등록부에 기억된 내용들은 /usr 구획에 보관되며 기타 모든 다른 자료들은 뿌리로 설정된 구획에 보관된다. 일정한 공간을 등록부로 제한하거나 허용하려면 그 등록부를 제한된 량으로 설치할수 있다. 표준웃준위 Red Hat Linux 등록부들의 기본역할은 다음과 같다.

/bin 표준체계편의프로그람들이 여기에 보관된다.
/boot 핵심부, 기동적재프로그람(boot loader), 기억기배치표들과 묘듈세부들이 여기에 보관된다. 보통 개별적인 구획에서 설정한다.
/dev 장치파일들이 여기에 보관된다.
/home 사용자의 홈등록부들이 여기에 보관된다.
/mnt /mnt/cdrom 과 /mnt/floppy 와 같이 림시로 설치한 파일체계들이 보통 여기에 보관된다.
/opt 선택적으로 설치된 프로그람들이 여기에 보관된다.
/root 뿌리사용자의 홈등록부이다. 이것은 설치된 뿌리구획(/)과 혼동되지 말아야 한다.
/sbin 표준체계관리파일들이 여기에 보관된다. 이 파일들은 뿌리사용자만이 호출할수 있다.
/usr 추가된 체계쏘프트웨어와 관리도구들이 여기에 보관된다.
/var Log 파일들과 인쇄자료들이 여기에 보관된다. 일반적으로 Log 파일들은 Web 싸이트로부터 Linux 콤퓨터의 다른 자료들을 다른 식으로 쉽게 밀어 낼수 있기때문에 Web 봉사기들의 제한된 공간에 개별적으로 설치한다.
/etc 관리파일들과 구성파일들이 여기에 보관된다. 이미전에 이 등록부에 있던 기타지령들을 /bin 혹은 /sbin 으로 옮긴다.

Device 이것은 매 구획에 대한 Linux 장치이름을 지정한다. IDE 디스크들에 대한 구동기들은 hdx 라고 표시한다. 여기서 x 는 구동기(a 는 첫번째 주구동기, b 는 첫번째 종속구동기, c 는 두번째 주구동기, d 는 두번째 종속구동기)를 지정하는 문자이다. 한개의 IDE 하드디스크가 있으면 그것은 디스크 hda 이다. SCSI 디스크는 sdx 로 표시되는데 x 는 다시 구동기를 나타낸다. 디스크의 구획들에는 hda1 로부터 시작하여 증가하는 번호를 붙인다. 일반적으로 구획 1～4 는 첫번째의 구획들이며 5 와 그이상의것들은 많은 DOS 체계에 있는 공통적인 론리구획들을 나타낸다. 그림 4-4 에서 기동등록부(hda1)와 뿌리등록부(hda5) 그리고 교환구획(hda6)과 결합된 Linux 구획들을 볼수 있다.

Requested 이것은 MB 단위로 구획의 최소크기를 나타낸다.

Actual 이것은 구획에 실제로 할당된 공간을 나타낸다. Disk Druid 는 하드디스크의 빈 공간을 쓸수 있게 하고 그 공간을 구획으로 다 리용함으로써 크기를 증가시키는 확장가능한 Linux 구획을 만든다. DOS 구획들은 현실적이면서도 필요한 값들과 정합되여야 한다.

Type 이 마당은 Disk Druid 로 작성할수 있는 구획의 류형을 나타낸다. 가능한 값들은 Linux native, Linux swap, Linux RAID, DOS 16-bit<M 과 DOS 16-bit>=M 이

다(여기서 FAT 혹은 NTFS 구획을 작성할수 없지만 Disk Druid 는 현재의 공통적인 Microsoft Windows 구획들을 인식한다.).

화면에서 다음으로 주요한 부분은 5 개로 된 과제단추묶음이다. 설치에 필요한 Linux 구획을 만들려면 Add 단추를 리용해야 하며 이미 있는 DOS 와 Windows 자료를 Linux 에서 호출할수 있는가 확인하거나 이전에 설정한것들을 변화시키기 위하여 Edit 단추를 리용할수 있다. Delete 단추로 개별적인 구획을 삭제할수 있다. 다시 시작하려면 Reset 단추를 리용해야 한다. 자료에 대하여 더 빠른 속도나 보안을 요구하면 Make RAID Device 단추는 독립적인 구획렬에 파일을 배포한다.

마지막으로 Drive Summary 부분이 있다. 이것은 체계의 매 하드디스크에 대하여 한 행씩 표시하며 장치들에 다음과 같은 정보를 포함한다.

Drive 이것은 앞에서 본 하드디스크구동기에 대한 이름이다.

Geometry 이것은 구동기에 대한 원기둥수와 자두수 및 분구수를 나타낸다(그 순서대로).

Total 이것은 디스크에서 총체적으로 리용할수 있는 공간을 MB 단위로 나타낸다.

Free 이것은 추가된 구획들에 리용할수 있는 구동기의 용량을 MB 단위로 나타낸다.

Used 구동기에서 리용된 총 공간을 MB 단위로 나타낸다. 이 수자는 실지 구획들로 할당된 용량가운데서 리용된 용량이 어느 정도인가를 나타낸다. 이 구획들은 다 차지 않았지만 남아 있는 공간은 다른 구획을 더 할당할 정도로 많지 못하다. Linux 의 주요구획들에 대한 번호는 령보다 커야 하며 1GB 이상의 구획이여야 한다. 이만한 공간이여야 Linux 를 설치할수 있다. 공간이 없다면 제 3 장에서 Linux 설치를 위한 공간을 어떻게 할당하여야 하는가를 서술한 부분을 참고할수 있다.

Used(%) 구획에 대하여 구동기에 할당된 공간의 비률을 나타낸다. 이 수자가 100%보다 작으면 구획을 더 추가할수 있다.

마지막으로 화면의 밑부분에 가로 놓인 단추띠에는 Hide Help 와 Back 그리고 Next 단추들이 있다. 페지의 왼쪽우에 있는 해설화면을 숨기려면 Hide Help 단추를 눌러야 한다. 이전의 차림표에로 돌아 가려면 Back 단추를 눌러야 한다. 다음차림표에로 이행하려면 Next 단추를 눌러야 한다. Next 단추는 Linux 에 설치될 구획들이 정확히 구성된후에만 리용할수 있다.

새 구획추가

새 구획을 추가하기 위하여 Disk Druid의 기본화면에 나타나는 Add단추를 누른다. 그러면 화면에는 다음과 같은 주요정보에 대한 질문이 현시된다.

- 설치위치(Mount point)
- MB 단위의 크기 (Size in megabytes)
- 할당되지 않은 디스크공간을 필요한만큼 리용하도록 구동기를 증가시킬수 있는가(Use Remaining Space 검색칸을 통해서)
- 구획형태(선택목록마당으로부터)
- 어느 구동기에서나 구획을 만들수 있다. 한개이상의 구동기들이 있고 허가된 모

든 구동기들이 구획을 만드는데 충분한 공간을 가지고 있으면 Disk Druid 는 어느 구동기를 리용하겠는가 결정할수 있다. 지정된 디스크에 구획을 만들려면 그 디스크가 검사되였는가를 확인해야 한다.

적어도 다음의 구획들을 만들려면 Add 단추를 리용해야 한다.

교환구획 Linux 는 교환할 때 리용할수 있는 독립적인 구획을 요구한다. 이것은 물리적인 RAM 을 전부 다 써 버리고 조작체계가 기능을 유지하기 위하여 가상기억기(RAM 으로 가상한 디스크공간)를 리용해야 하는 경우에 필요하다. 일반적으로 제한조건은 그 크기가 최소한 체계가 가지고 있는 물리적인 RAM 용량과 정합되여야 한다는것이다. 즉 하드디스크에 충분히 빈 공간이 있으면 RAM 의 물리적인 크기의 2 배만한 교환구획을 만든다. 따라서 MB 체계에 MB 와 64MB 사이 교환구획을 만든다. 구획형태를 Linux 교환으로 설정하며 설치위치를 나타내지 말아야 한다. 기타 구획들을 위한 공간을 보관하려면 Use Remaining Space 검색칸을 선택하지 말아야 한다.

주의 Red Hat Linux 7.1 에서 교환구획들은 최소한 MB 용량을 요구한다. 워크스테이션 혹은 무료형부류의 설치를 선택했다면 교환구획에 Red Hat 7.1 설치프로그람은 64MB 를 할당한다. 그러나 봉사기부류의 설치를 선택했다면 250MB 의 교환구획을 얻게 된다.

뿌리구획 이 장에서는 두개의 구획에 Linux 를 설치하려고 한다(다음장에서 Linux 를 설치하기 위한 다중구획들 혹은 다중구동기들의 리용을 고찰한다.). 그러자면 또 하나의 구획을 추가해야 한다. 일단 Linux 가 설치되면 작업할 공간으로 적어도 1GB 의 구획이 있어야 한다. 이 구획은 Linux Native 형이여야 하며 </설치위치>를 가져야 한다. 그리고 초기크기보다 구획이 더 크게 증가할 때 리용하려는 구동기에 할당되지 않은 빈 공간이 있으면 Use Remaining Space 가 선택되여야 한다.

기동구획 Red Hat Linux 7.1 구획이 구동기에 자동적으로 추가되면 기정대로 기동구획이 설정된다. 기동구획들에는 Linux 를 기동하는데 필요한 주요요소들 즉 핵심부, 체계배치표, 하드웨어모형위치들이 들어 있다. 대부분의 Linux 체계들에서 기동구획은 하드구동기의 1024 번째 원기둥에 있어야 한다. 그러나 fdisk 로 구획을 설정했으면 Red Hat Linux 7.1 은 후에 LILO 를 설정할 때 핵심부파라메터로 1ba 을 추가하여 이 제한을 무시하도록 한다.

구획편집

현행 구획목록으로부터 구획들을 선택하여 이미 있는 구획을 편집할수 있는데 이를 위해서 Edit 단추를 찰칵한다. 그러면 새 구획을 추가할 때 리용했던것과 같은 창문이 나타나는데 다만 여기의 모든 마당들이 편집하고 있는 구획설정들과 일치하는 마당이라는것만 다르다.

초기부터 존재하는 DOS 구획들이 있으면 그것들에 설치위치를 지정하여 Linux 에서 리용할수 있게 한다. 이를 위해서 Linux 에서 호출할 구획을 선택하고 Edit 단추를 누른다음 그 구획에 대한 설치위치를 기입한다.

실례로 한개의 DOS 구획이 있으면 /dos 로 설정할수 있다. DOS 와 Windows 의 C 와 D 구동기들인 두개의 DOS 구획들이 있으면 설치위치로 /dosc 와 /dosd(혹은 /c 와 /d)를 설정할수 있다.

구획삭제

Linux 교환구획들과 뿌리구획들을 만드는 과정에 잘못한것이 있으면(혹시 교환구획이 너무 크거나 뿌리구획이 너무 작아 질수 있다.) 그 구획들을 삭제하고 다시 추가할수 있다. 이를 위해서 현행 구획목록에서 그 구획을 선택한 다음 Delete 단추를 누른다.

경고	구획을 삭제할 때 주의해야 한다. Linux 설치를 시작하기전에 이미 존재한 구획과 보관하려는 중요한 자료들과 쏘프트웨어를 포함한 구획들을 실수하여 삭제하지 말아야 한다.

재설정

Linux 구획들을 만드는 과정에 다시 시작할 필요를 느끼면 Reset 단추를 찰칵한다. 그러면 Disk Druid 를 시작하기전 상태에 대한 구획표를 다시 보관한다.

RAID 장치만들기

Linux 를 봉사기로 설치하려면 다중사용자를 위한 콤퓨터를 설정해야 한다. 봉사기에서 프로그람과 자료들은 매우 중요한것으로 된다. 저가격디스크의 여벌배렬(RAIDs:Redundant Array of Independent Disks)로 알려 진 다중하드구동기들에 자료를 분산시켜 그것을 보호할수 있다. RAID 계렬구획들을 설정하려면 Make RAID Device 항목으로 그것들을 구성할수 있다.

이동

일단 구획만들기와 구성이 끝났으면 이동을 위한 준비가 되였다고 볼수 있다. Next 단추를 눌러서 이것을 수행할수 있다.

Linux 구획양식화

다음단계는 Linux 를 설치하기 위한 준비로서 Linux 구획을 양식화(format)하는것이다. Linux Native 형식의 구획목록은 그것들이 가지고 있는 설치위치들에 대한 표시와 함께 나타난다. 구동기를 지정하는 검색칸을 선택하여 새로 만든 구획들을 양식화하는데로 넘어 가야 한다.

Linux Native 형식의 추가구획들이 있는데 그것들이 새 구획들이라면 그 구획들을 양식화해야 한다. 이미 구획들(새로운 Red Hat 리용자들을 위한것이 아닌)이 존재하고 있고 그 구획들에 자료들이 있으면 양식화하는 구획들을 표시하지 말아야 한다.

또한 양식화하는 과정에 불량한 블로크검사를 진행하도록 지시할수도 있다. 이것은 특히 새로운 하드디스크구동기를 위해서도 좋다.

LILO 설정

LILO 는 Linux 의 초기적재프로그람이다. 체계가 정확하게 기동되도록 하기 위해서는 LILO 를 구성하고 설치해야 한다. LILO 는 또한 기동시에 Linux 는 물론 존재하고 있는 Windows 나 DOS 체계를 호출하도록 하면서 조작체계의 기동을 선택하게 할수 있는 2 중기동특성(dual-boot features)을 제공한다.

그림 4-5 에서 보여 주는것처럼 이 처리에는 4 개의 부분들이 있다. 첫째로 하드구동기 혹은 기동디스크에 LILO 를 설치하기 위한 위치를 결정해야 한다.

하드구동기에 LILO 를 설치할수 있는 두 위치를 선택할수 있다. 즉 주기동레코드(MBR :Master Boot Record) 혹은 기동구획의 첫 구역위치들이다. OS/2 나 Windows NT/2000 과 같은 초기적재프로그람으로 조작체계를 실행하지 않거나 또는 Partition Magic 나 System Commander 와 같은 다른 초기적재프로그람들을 설정하지 않는 경우에는 MBR 에 LILO 를 설치해야 한다. 이미 이것들중 어느 하나에 초기적재프로그람이 있으면 그 경우에는 LILO 가 필요 없다. 그러므로 Linux 기동구획에 LILO 를 설치해야 한다.

주의 Linux 체계를 기동시키려면 이 단계에서 기동디스크를 만들든가 LILO 를 설치해야 한다. 초기설치시에 반드시 LILO 를 설치해야 한다.

대부분의 하드구동기들은 선형방식으로 호출될수 있다. 하드구동기들이 론리블로크주소화(LBA)방식(BIOS 를 검사)을 리용하지 않는다면 선택된 Use Linear Mode 항목을 보존해야 한다.

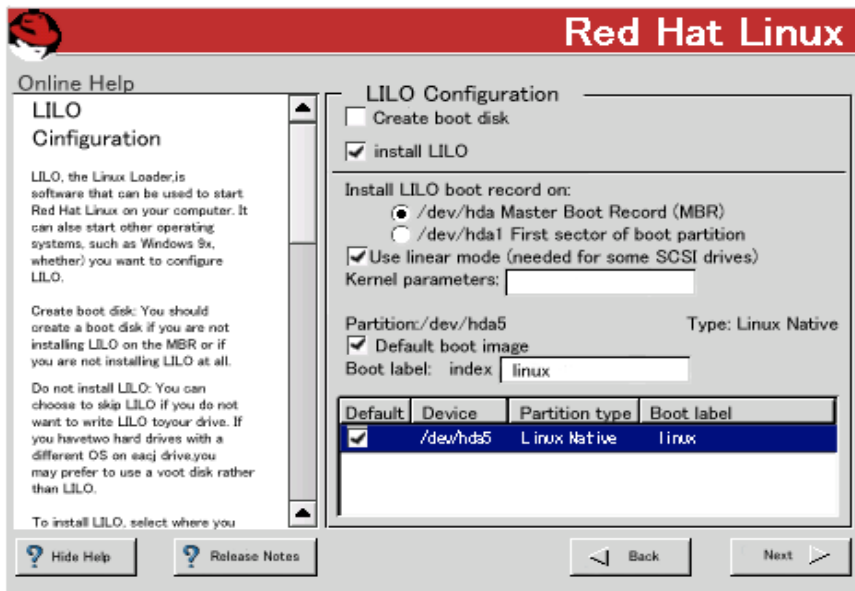



그림 4-5. Linux 기동적재기의 구성

화면의 다음부분에서 조작체계들이 있는 모든 구획들을 볼수 있다. Default Boot Image 를 설정할 때에는 콤퓨터를 기동할 때 기정대로 기동하는 조작체계(보통 Linux

혹은 Microsoft Windows)를 설정한다. Boot Label은 콤퓨터가 기동할 때 어느 조작체계로 기동하겠는가 선택하는데 리용되는 이름이다(그림 4-5에서는 linux).

망구성

구성하려는 다음의 항목은 망이다. 이 장에서는 표준체계설치를 하기때문에 망구성 설정을 선택하지 않는다. 제 27장에서 망구성에 대하여 더 상세하게 설명한다.

방화벽구성

다음의 항목은 방화벽을 구성하는것이다. 이 장에서는 독립적으로 체계를 설치하기 때문에 방화벽구성를 선택하지 않는다. 제 31장에서 더 상세하게 방화벽구성에 대해서 고찰한다.

언어지원선택

여기서는 설치가 완성되여 끝난후에 Red Hat Linux 체계에서 자기가 리용할 언어를 설정할수 있다. Red Hat Linux 7.1에서는 104개의 각이한 언어들 혹은 방언들을 선택할수 있다. 디스크공간이 부족하면 리용하려는 언어만을 선택해야 한다. 즉 104개의 모든 항목을 선택하려면 135MB의 디스크공간이 있어야 한다.

시간지대구성

다음단계는 콤퓨터시계와 시간지대를 구성하는것이다. 두가지를 결정해야 한다.

첫째로, 국부시간구역 혹은 프랑스의 아크로님(acronym)에 대등한 만능시간(UTC:Coordinated Universal Time)으로 알려 진 그리니치시간(GMT 즉 Greenwich Mean Time)의 어느 하나로 체계의 시계를 설정하도록 지시해야 한다. 콤퓨터에서 Linux만 실행하려면 UTC로 시간을 설정한다. 그러면 Linux는 시간지대에 대한 현재시간으로 그것을 변환한다. 그러나 PC에서 다른 조작체계를 실행한다면 이 항목을 선택할수 없다.

둘째로, 리용할수 있는 시간지대목록으로부터 실지 시간지대를 선택해야 한다.

회계구성

다중사용자조작체계인 Linux는 조작체계를 리용하기 위해서 적어도 한명의 사용자가 있어야 한다. 모든 체계들에서는 뿌리사용자가 있어야 하므로 뿌리통과암호를 제공하도록(확인을 위하여 두번) 재촉문을 제시한다. 뿌리사용자는 전반적인 강력한 체계관리기이다. 뿌리사용자로 가입되면 모든 사용자파일들을 볼수 있으며 체계관리의 모든 과제들을 수행할수 있다. 그리고 원한다면 체계에 있는 모든 파일들을 삭제할수도 있다. 이것은 강력한 회계이므로 다른 사람이 자기의 콤퓨터체계에 접근하려고 할 때 이 통과암호를 안전하에 보존해야 한다.

다음에 필요한 회계이름과 통과암호에 기초하여 개별적인 회계들을 설정할수 있다. 이를 위해서 Full Name 본문칸에 완전한 이름을 입력한다.

인증구성

다음으로 인증구성을 선택하는 재촉문이 나타난다. 이것은 5 개의 항목으로 이루어져 있으며 개개는 개별적으로 선택될수 있다. 그것들은 서로 배척하지 않는다. 그 항목들은 다음과 같다.

Enable MD5 Passwords 전형적으로 Unix 체계들에서는 통과암호를 보관하기 위하여 비교적 약한 암호화방식를 리용하였다. 이 항목에서는 Linux 의 사용자통과암호들을 보관하도록 하는 더 엄격한 암호화방식들을 리용할수 있다. 이 항목은 기정으로 선택되여 있다.

Use Shadow Passwords 그림자통과암호들을 리용하는것은 공격자 혹은 정해 진 체계사용자들이 여가시간에 사용자자료기지를 도적질한후 체계관리통과암호를 깨뜨리려고 하는것을 더 어렵게 하기 위하여 설계된 기술이다. 이 항목은 기정으로 선택되여 있지 않다.

Enable NIS 이것은 많은 Unix 망들 특히 Sun Solaris 에 기초한 봉사기망들에서 일반적으로 쓰는 망인증형식이다. 망에 있는 콤퓨터집합의 령역이름을 지정하거나 NIS 사용자이름들과 통과암호들이 보관되여 있는 특수한 봉사기를 지정할수 있다. 기정으로 이 항목은 선택되여 있지 않다.

Enable LDAP 라이트급등록부호출규약(LDAP:Lightweight Directory Access Protocol)은 사용자의 특별등록부들에 설정된다. 망에 있는 LDAP 봉사기의 이름을 알면 LDAP 를 리용할수 있다. 기정으로 이 항목은 선택되여 있지 않다.

Enable Kerberos Kerberos 는 망을 통하여 전송되는 통과암호를 암호화하는 안전수단이다. Kerberos 는 암호화방식을 포함하고 있는 kerberos 자격증을 주는 봉사기를 호출할것을 요구한다. 이 통신규약을 리용하는 모든 Linux 봉사들을 설정하지 않고서는 Kerberos 를 리용할수 없다. 이 항목은 기정으로 선택되여 있지 않다.

이러한 항목들에 익숙되지 않고 구태여 그것들을 변경할 리유가 없다면 기정선택으로 넘어 가야 한다.

프로그람(package)의 선택

이제는 하드디스크가 구성되고 Linux 구획이 양식화되였으므로 실제로 프로그람체계를 설치할수 있는 준비가 되였다고 볼수 있다. 기정설치에는 필요한 모든 핵심적인 쏘프트웨어들이 들어 있지만 그림 4-6 에서 보여 준것들을 비롯하여 여러개의 선택요소들을 리용할수 있다.

매 요소들은 전화접속방식, Web 항행 등과 같이 특수한 과제들과 관계되는 프로그람묶음이다. 개별적인 검색칸에 표식을 설정하여 개별적으로 프로그람들을 선택할수 있으며 목록안에 있는 모든 항목들을 선택할수 있다.

목록마당아래에 나타나는 Select Individual Packages 항목은 매 요소에 속한 개별적인 프로그람묶음을 선택하려고 할 때 리용할수 있다. 그러나 처음으로 Linux 를 리용하는 사람이라면 프로그람묶음을 선택하기가 힘들다. 이 선택항목들을 선택하지 않고 넘어 가면 매 요소가 전부 설치된다는것을 알아야 한다.

충분한 디스크용량(3GB 이상, 4GB 이면 적당하다.)이 있다면 완전한 Linux 를 설치할수 있는 모든것을 다 선택할수 있다. 그림 4-6 의 아래에 보여 준 총 설치크기에 주목을 돌려야 한다. 제공되는 공간이 작으면 최소한도로 다음의 설치요소들을 고려해야 한다.

그림 4-6. 추가요소의 선택

- Printer Support(인쇄기가 없으면 필요 없다.)
- X Windows System
- GNOME
- Mail/WWW/News Tools
- Dos/Windows Connectivity
- Multimedia Support

충분한 구역이 있으면 제 11 장 《KDE 의 리용》을 방조하기 위해 GNOME 과 KDE 를 설치할수 있다. X Windows 설치를 선택한 다음 비데오기판과 현시장치를 구성하는데 착수한다. 그렇지 않으면 다음절의 마감부분에서 보게 되는것처럼 직접 설치할수 있다.

주의 보여 준 최소한도의 요소들보다 더 많은것을 설치하려면 두개의 Red Hat Linux 7.1 설치 CD-ROM 이 있어야 한다. 두번째 설치 CD-ROM 은 이 책과 함께 제공되지 않았기때문에 www.redhat.com 에서 공적인 CD-ROM 을 구입하든가 www.cheapbytes.com이나 www.linuxmall.com 과 같은 대리공급자로부터 CD-ROM 그자체를 직접 구입해야 한다.

X Windows 구성

X Windows 는 Linux 의 도형사용자대면부(GUI)이다. 제 7 장 《X Windows 의 설치와 구성》에서는 X Windows 구성에 대하여 자세히 고찰한다. 이제는 구성프로세스를 빨리 실행해 보자. 구성은 매우 복잡해서 장전체에서 그것을 반영할수 없으므로 일부 문제들은 뒤장에서 해결하기로 한다.

우선 현시장치형태목록이 제시된다. 현재의 현시장치와 정확히 맞는것을 찾으면 그것을 선택해야 한다. 그렇지 않으면 Unprobed Monitor 을 선택하거나 제 7 장을 고찰할 때까지 기다려야 한다. 정확치 못한 구성은 현시장치에 손해를 줄수 있으므로 정확하게 구성해야 한다.

다음으로 가지고 있는 영상기판의 형태가 무엇인가를 결정해야 한다. 이것이 실패하면 리용할수 있는 영상기판의 목록이 제시된다. 가지고 있는 기판들중에서 맞는것을 하나 선택한다(맞지 않으면 Generic Standand VGA 를 선택하거나 제 7 장에서 X Windows 를 구성할 때까지 기다려야 한다.). 다음 영상기판의 비데오기억기크기를 선택한다. 이 정보들에 대하여서는 영상기판문서를 참고할수 있다. 일단 선택하였으면 선택된 설정들이 체계에서 작업하는가를 확인하기 위해 Test This Configuration 을 찰칵할수 있다.

Customize X Configuration 항목을 선택하면 현시장치(aka Video Modes)에 대한 X Windows 의 해상도를 선택할수 있다. Use Graphical Login 항목으로 도형방식의 가입화면을 설정할수 있다. Linux 을 설치한후에 도형방식으로 구성하려면 Skip X Configuration 항목을 선택해야 한다. 그러면 도형방식의 구성을 제 7 장으로 넘길수 있다.

주의

Red Hat 의 판본들을 차례로 포함하고 있는 일부 배포물들에서는 박자소편들의 목록을 제시한다. 기판에 박자소편이 없거나 그것이 무엇을 하는것인지 모르면 No Clockchip 를 선택해야 한다. 어림짐작하지 말아야 한다. 기판의 박자소편에 대하여 모른다면 제 7 장에서 고찰할 때까지 이 작업을 하지 말아야 한다.

지금까지의 고찰로부터 기본적으로 구성에 대한 요구가 만족되였다고 할수 있다. Next 를 누르면 실제적인 설치가 진행된다는것을 통보한다. 대부분의 배포물들은 설치과정에 대한 기록파일을 만든다. Red Hat Linux 7.1 은 이것을 /tmp/install.log 파일에 보관한다. 다른 여러가지 배포물들은 비슷한 파일로 자기의 설치기록파일을 만든다. Next 를 찰칵하면 설치가 시작된다.

설치과정

설치는 자동적으로 진행되며 이것이 처리되는동안에 입구파라메터는 아무것도 요구하지 않는다. 화면에는 설치가 끝나자면 얼마나 오래 기다려야 하는가를 알려 주는 시계가 나타난다. 쏘프트웨어를 설치하는동안 화면에는 현재 어느 프로그람묶음이 설치되여있고 얼마만한 프로그람묶음과 몇 MB 가 남아 있는가 그리고 설치에 대한 전반적인 진척과정을 나타내는 설치의 현재상태와 전체 프로그람들이 현시된다.

기동디스크만들기

Linux 체계가 성과적으로 설치되였다면 설치의 마지막단계로 진입할수 있다. 즉 기동디스크를 만들어야 한다. 기동디스크를 만드는것은 매우 현명한 일이다. 왜냐하면 기동디스크는 비상사태가 일어 난 경우에 체계를 기동하도록 하기때문이다. 잘못된 체계를회

복시키려고 할수 있다. 이를 위해서 그림 4-7에서 지적된것처럼 빈 플로피디스크(양식화된것)를 끼워 넣고 Next를 찰칵한다.

Red Hat Linux 7.1 설치프로그람은 플로피디스크에 기동파일을 복사한다. 이 공정이 완성되면 설치프로그람은 완료된다.

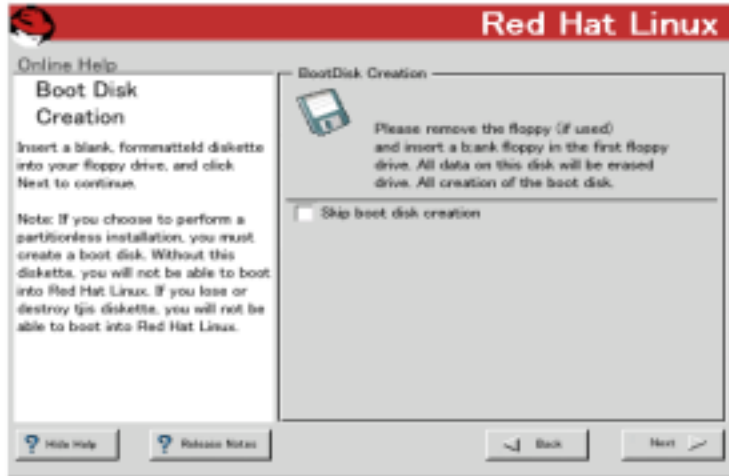



그림 4-7. 기동디스크만들기

체계의 재기동

마지막으로 콤퓨터로부터 플로피디스크를 꺼내고 재기동하라는 재촉문이 나타난다. 일부 Linux배포물들은 CD-ROM을 자동적으로 꺼낸다. 이 과정이 일어 나지 않으면 재기동하기전에 CD-ROM을 꺼내야 한다. Linux를 기정조작체계로 해두면 체계는 기동과정에 교대조작체계를 선택해 주기 바라면서 도형방식의 재촉문상태에서 5초동안 있다가 Linux로 기동한다.

Red Hat Linux 7.1에서는 콤퓨터에 한개이상의 조작체계가 있으면 조작체계를 선정하기 위하여 아래우방향건(↓,↑)을 리용한다. Linux를 선택하고 모든 처리가 다 끝나면 Red Hat Linux는 가입등록재촉문을 제시한다. 초기에 제공한 사용자이름 root와 통과암호를 리용하여 뿌리사용자로 여기에 가입할수 있다. 지령재촉문은 다음과 같다.

[armand @ localhost armand]$

후에 지령들에 대해서 배우게 되겠지만 그 리용법은 간단하다. 지령을 입력하고 그것을 실행시키기 위하여 Enter 건을 누른다. 가입에서 탈퇴하려면 **exit**를 입력하고 Enter를 눌러야 한다.

Linux는 초기문자방식일 때도 한번이상 가입할수 있다. Alt+f1∼Alt+f6을 결합하면서 6개의 가상적인 조종탁(console)사이를 절환할수 있다. 매개의 가상조종탁에 개별적으로 가입해야 한다. 여러 사용자들로 가입할수 있으며 기타 과제들을 수행하거나 다른 가상조종탁에 같은 사용자로 가입할수도 있다. 즉 이것은 유연한 작업환경들이 정합되게 하는 Unix와 Linux특성들중의 하나이다.

▶미리 알아두기

이제는 Linux 체계를 실행시키는 방법을 알게 되였다. 이 장의 단계들을 거쳐서도 Linux 체계를 실행하기 힘든 특수한 형편에 있는 사람이라면 특수한 경우의 설치를 설명한 제 5 장 《특수설치》를 보아야 한다. 즉

- 하드구동기의 구획에 설치(Linux 를 CD-ROM 구동기로 설치할수 없는 경우에 이것은 매우 중요하다.)
- 여러 구획들 혹은 하드디스크들에 설치(이것은 설치에 필요한 디스크공간이 여러개의 디스크에 분산되여 있거나 이 방식으로 Linux 를 분할하는것이 편리할 할 때 매우 쓸모 있다.)
- 구획을 구성하는 Disk Druid 대신에 fdisk 를 리용(Disk Druid 는 리용할수 있지만 Linux 를 위한 표준은 아니다. 즉 fdisk 를 리용하면 이 표준도구의 리용법을 알수 있으며 또 전문가사용자가 요구하는 강력한 도구들을 호출할수 있다.)

이러한 특수설치들이 필요하다면 제 5 장으로 넘어 가야 한다. 체계가 정확히 기동했으면 곧장 X Windows 배우기로 넘어 갈수 있다. 제 6 장 《X Windows 의 개괄》에로 넘어 가 계속할수 있다.

제 5 장. 특수설치

제 4 장 《Red Hat Linux 7.1 의 설치》에서는 국부적인 CD-ROM 으로 한개의 하드디스크구획에 Linux 를 설치하는 간단한 방법을 보았다.

이 장에서는 일반성이 없는 몇가지 변종의 설치방법을 본다. 그중 첫번째는 하드디스크구획에 설치하는것이다. 이것들은 국부적인 CD-ROM 으로부터의 설치를 실현할수 없는 경우에 해당된다. 실례로 Red Hat 설치쏘프트웨어가 몇개의 CD-ROM 구동기중에서 하나라도 인식할수 없거나 CD-ROM 구동기를 림시적으로만 호출할수 있는 경우이다. 이러한 경우에 한가지 방도는 하드구동기의 빈 공간에 Red Hat CD-ROM 의 내용을 복사하고 다음 그 하드구동기로 그것을 설치하는것이다.

한편 오늘날 값이 눅은 하드구동기를 가지고서도 Linux 를 설치할수 있는 쓸모 있는 여러개의 구획이나 하드구동기들을 마련해야 한다. 이러한 경우에 여러개의 구획과 구동기에 Linux 를 전개함으로써 체계의 성능을 최대화할수 있다.

마지막으로 이 장에서는 하드디스크에 구획을 설정하는 Disk Druid 대신에 fdisk 프로그람을 리용하는 방법을 주었다. fdisk 는 Linux 의 모든 배포물에서 쓸수 있도록 표준화되였지만 Disk Druid 는 Red Hat Linux 에서만 리용할수 있다.

하드디스크구획에 의한 설치

일부 사용자들은 Linux 를 실행하기 위해서 하나의 구획이 아니라 서로 다른 하드디스크구획에 Linux 를 설치해야 하는 어쩔수 없는 경우에 부닥칠수 있다.

가능한 경우는 다음과 같다.

- Linux 설치프로그람이 CD-ROM 구동기를 인식하지 못하는 경우
- 항시적으로 CD-ROM 구동기가 부족하지만 CD-ROM 의 내용을 하드디스크에 복사하기 위하여 CD-ROM 을 호출할수 있는 경우

- CD-ROM 구동기가 있지만 절환할수 있는 플로피디스크와 CD-ROM 구동기로 기동될수 없는 휴대형콤퓨터를 가지고 있는 경우

이 모든 경우에 하드디스크구획에 설치하는 과정은 기본적으로 같다.

1. 준비된 하드디스크구획에 Red Hat CD-ROM 으로부터 ISO 디스크영상을 만든다. 두개의 Red Hat CD-ROM 을 복사하면 그것들은 같은 등록부에 정확히 복사된다.
2. 반드시 정확한 설치플로피디스크를 준비해야 한다.
3. boot.img 파일로 만든 설치플로피디스크로 기동하여 설치처리를 시작한다.

주의 Linuxconf 와 같은 나머지프로그람을 설치하려면 두개의 Red Hat 설치 CD 가 있어야 한다. 많은 주요콤퓨터기억기들이나 www.redhat.com에서 직접 완전한 Red Hat Linux 7.1 프로그람묶음을 구입할수도 있다. 또는 직결해서 CD 들을 적재할수 있다. 또한 Cheap Bytes(www.cheapbytes.com)나 Linux Mall (www.linuxmall. com)과 같은 판매자들로부터 내리적재한 CD-ROM 을 구입할수 있다.

하드디스크구획에 CD-ROM 을 복사

하드디스크구획으로 설치하기 위해서는 Linux 설치 CD-ROM 의 내용을 복사하는데 전적으로 쓸수 있는 구획이 있어야 한다. 이 구획에는 표준 Red Hat Linux 7.1 설치 CD-ROM 의 내용을 전부 복사할수 있도록 최소한 650MB 의 빈 공간이 있어야 한다. 표준프로그람묶음보다 더 많이 설치하려면 이 구획은 2배나 더 큰 공간을 요구한다.

Red Hat Linux CD-ROM 으로부터 ISO 영상파일을 만들기 위하여서는 cd1.iso 와 같은 파일에 있는 CD-ROM 의 내용들로부터 프로그람을 만들어야 한다. 이것을 위한 한가지 방법은 처음에 CD-ROM 을 설치하고 다음에 ISO 영상을 만드는 뿌리사용자방식으로 다음의 지령들을 실행하는것이다.

```
# mount -t iso9660 /dev/cdrom /mnt/cdrom
# mkisofs -J -r -T -o /tmp/cd1.iso /mnt/cdrom
```

주의 /tmp 또는 /home/mj 와 같은 보조등록부에 CD ISO 영상파일을 배치한다. 이것이 있으면 설치프로그람은 그 영상의 위치를 쉽게 검색한다. 두개의 설치 CD-ROM 에 대한 ISO 영상을 만들려면 그것들을 같은 등록부에 배치해야 한다.

CD-ROM 구동기들이 여러개 있으면 /dev/cdrom 으로 바꾼다. mkisofs 지령에서 -o 스위치는 만들려는 ISO 영상의 파일이름앞에 온다. -T 스위치는 MS-DOS 와 같이 긴 파일이름정보들을 조종할수 없는 조작체계들에서 그것들을 보존하는 매 등록부에 TRANS.TBL 파일을 추가한다. -r 스위치는 긴 파일이름을 승인하는 《Rock Ridge》확장자를 리용하며 -J 스위치는 Microsoft Windows 콤퓨터에서 이 ISO 를 리용할수 있는 《Joliet》레코드들을 리용한다.

두번째 Red Hat Linux 7.1 설치 CD-ROM 이 있으면 같은 등록부에 두번째 ISO 영상을 만들고 싶을것이다.

주의 이 책에서 설명하는 Red Hat Linux 7.1 판본에는 한개의 설치 CD-ROM 만이 제공되였다.

그 크기때문에 ISO를 만드는데는 몇분이 잘 걸린다. cd1.iso파일에 대한 결과를 검사하기 위하여서는 정규적인 CD-ROM 을 설치하는 방법으로 그 파일을 설치할수 있다. 즉

mount -t iso9660 -r -o loop /tmp/RedHat/*cd1*.iso /mut/cdrom

그 결과는 정규적인 CD-ROM 을 설치한것과 꼭 같아야 한다. 이 경우에 Red Hat Linux 7.1설치 CD-ROM의 파일들과 ISO에 있는 파일들을 비교할수 있다.

설치플로피디스크준비하기

제3장 《Red Hat Linux 7.1의 설치준비》에서는 필요한 설치플로피디스크들을 만드는 방법을 보았다. 상세한것에 대하여서는 그 장에 있는 《플로피디스크와 CD-ROM에 의한 설치》라는 절을 참고할수 있다.

일부 배포물들을 하드디스크구획으로 설치하기 위하여서는 기본기동설치플로피디스크와 구동프로그람 둘다 혹은 보조플로피디스크가 있어야 한다. 그러므로 설치처리를 시작하기전에 준비된 알맞는 디스크들을 가지고 있는것이 아주 좋다.

실제적인 설치과정

설치를 시작하기 위하여 기동플로피디스크를 기동시킨다. CD-ROM으로 표준설치할 때 리용한 단계(제4장 《Red Hat Linux 7.1의 설치》에서 서술)를 거쳐야 한다. 기동디스크는 제4장에서 보여 준 표준설치화면보다 적게 질문하면서 실지로는 도형방식화면인 본문방식화면을 제시한다. 설치언어와 건반을 선택한후에 설치방법화면이 나타난다. 이 화면에서 Local CD-ROM이 아니라 Hard Drive를 선택한다.

하드구동기를 설정한후에 그림 5-1 에서 보여 준것처럼 Linux 하드디스크영상을 가진 구획과 등록부를 지정하라는 재촉문이 나타난다. 보다 앞서 보여 준 실례에 기초하여 Directory Holding Images 본문창문에 /tmp를 입력할수 있다.

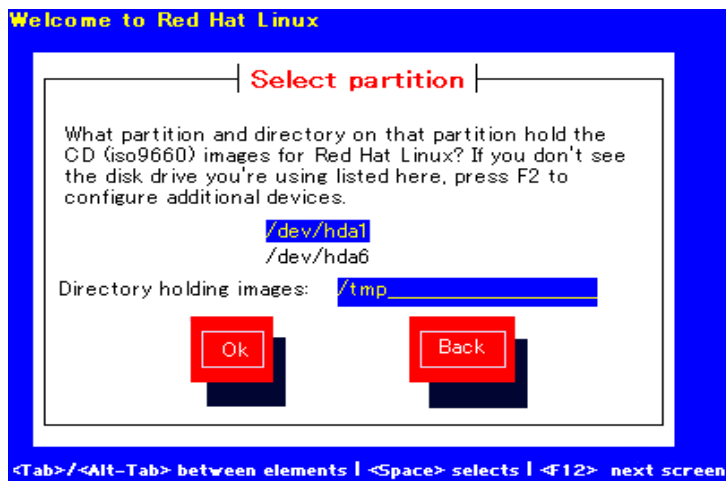



그림 5-1. 구획과 등록부의 선택

경고 이 책에서 쓴것처럼 이러한 지령들은 Red Hat Linux 7.1 의 문서에서 보여준것과는 조금씩 다르다. 여기서 읽고 있는것들은 내리적재된 이 조작체계의 판본에 대치되게 검사된것들이다.

주의 디스크구획을 위한 Linux 장치이름의 론의에 대해서는 제 4 장의 뒤부분을 참고할수 있다. 어느 구획에 Linux 설치영상이 포함되여 있는지 모르면 차례로 매 구획을 조사해 보아야 한다. 잘못 선택하였다면 그림 5-1 에서 보여 준 화면에 재촉문들이 나타나는데 이에 대해서 다시 조사해 볼수 있다.

나머지설치는 제 4 장에서 설명한것과 거의 같은 방법으로 진행하여야 한다.

다중구획에로의 Linux 의 설치

이 절의 제목은 무엇인가 잘못되였다. 결국 대표적인 Red Hat Linux 7.1 을 설치하기 위하여 적어도 3 개의 구획들 즉 기동파일을 위한 구획, 나머지조작체계를 보관하기 위한 구획 그리고 교환공간을 위한 구획을 리용해야 한다. Linux 를 보관하기 위하여 한개이상의 구획을 리용해야 하는 어쩔수 없는 경우가 있을수 있다.

물리적으로 린접하지 않은 두개의 유용한 구획들이 있는 디스크를 가지고 있는것이 명백한 하나의 리유이다. 두개의 서로 다른 물리적인 구획으로부터 하나의 린접한 기록집을 만드는것은 어렵다. 이 경우에는 보통 두개의 구획으로 Linux 설치를 분할하는것이 더 좋다. 실례로 여러개의 구획에 Linux 를 설치하면 속도와 자료유연성을 증가시킬수 있다. 한개이상의 디스크로 Linux 를 설치하기 위한 우점을 얻는데는 여러가지 방법들이 있다.

분할된 물리장치에 교환구획을 정하기

한개의 구동기에 큰 구획이 있고 두번째 물리구동기에 작은 구획(~200MB 범위)이 있을 때 두번째 구동기에 교환구획을 정하는것을 고찰해 보고 싶을수 있다. 자체교환이 자주 요구되면 이것은 현저하게 작업을 개선할수 있다.

이 스크립트를 고찰하자. Linux 설치와 교환구획은 같은 물리구동기에 있다. 많은 응용프로그람들이 실행되고 체계는 이미 얼마간 교환되였다. 새로운 응용프로그람을 기동하려면 기동되는 순서로 디스크로부터 기억기에 적재되여야 하며 동시에 기억기에 있는 자료는 디스크에로 교환되여야 한다. 구동기가 한번에 하나의 조작만을 수행할수 있는데 조작체계가 거의 같은 시간에 두개의 동작을 수행하게 함으로써 한개의 물리구동기는 갑자기 난관에 부닥치게 된다.

그러나 교환구획이 개별적인 물리구동기에 있으면 난관은 줄어 든다. 하나의 디스크는 다른 디스크가 기억기로부터 디스크로 자료를 옮기는 처리를 끝내는 동안 디스크로부터 기억기에로 넘어 가는 자료를 읽을수 있다. 지어 콤퓨터가 두개의 지령을 동시에 집행할수 없다고 해도 속도가 뜬 디스크조작이 두개의 디스크에서 전개되므로 CPU 는 디스크가 작업을 끝낼 때까지 기다리는 시간을 최소로 되게 한다.

주의 두개의 디스크구동기가 물리적으로 분리되여 있으면 그것들은 둘다 콤퓨터의 나머지기억기부분에 동시에 자료를 보낼수 있다. 매개 구동기를 물리적으로 자체의 디스크조종기에 접속함으로써 자료난관을 훨씬 줄일수 있다.

Linux를 여러개의 구획으로 분할

Linux 쏘프트웨어를 위한 두개이상의 구획들을 리용하는 한가지 방법은 론리적인 방법으로 Linux 쏘프트웨어의 기억기를 구획들로 분할하는것이다. 이것은 두가지 우점을 가진다. 즉

- /home 등록부나무와 같은 림계 Linux 등록부나무들로 쓸수 있는 디스크공간을 확장
- 유용한 구획들이 하나이상의 디스크에 있으면 여러개의 하드디스크에 디스크호출을 분할하여 작업을 개선

Linux 배포물을 두개의 구획으로 분할하는 방법을 고찰해 보자. 제 4 장에서 언급된것처럼 현존 DOS 구획에 대한 설치위치를 지정하는 방법의 배포물들에서는 지정된 설치위치로 추가적인 디스크들을 리용한다.

례를 들어 기본 Linux 구획에 대한 설치위치를 /으로 하는 경우 분할된 구획에 모든 사용자들의 홈등록부들을 보관하려면 그 구획에 대한 설치위치를 /home 으로 만들어야 한다. 그러면 /home 에 있는 임의의 보조등록부로부터 읽거나 쓸 때 실제로 /home 밖의 다른 등록부나무를 호출할 때보다 많은 구획이 호출된다.

Linux 설치를 두개의 구획으로 분할하는 방법에는 여러가지 다른 대중적인 방법들이 있다. 즉

- 많은 쏘프트웨어(문서처리기, Web 열람기, Windows 모방쏘프트웨어 같은 업무적인 쏘프트웨어를 포함)를 설치하면 /opt 나무등록부에 대부분의 쏘프트웨어들이 설치된다는것을 알수 있다. 기본Linux구획이 조작체계와 사용자자료를 포함할만한 충분한 구역을 가지고 있다면 추가구획을 /opt 에 설치하는것을 고려해 보아야 한다. 이것은 모든 응용프로그람들에 분리된 구역들을 주며 응용프로그람들이 실행될 때 자료를 호출하며 Linux 가 항상 실행하는 체계편의프로그람들과 배경과제들을 실행하기 위해 기본체계디스크는 여전히 비여 있게 한다.
- 많은 사용자들이 있거나 사용자의 홈등록부들에 다량의 자료를 보관하려면 추가적인 구획을 /home에 설치하는것을 고려해 보아야 한다. 이 방법으로 사용자디스크리용상태를 따로따로 현시할수 있으며 다른 사용자들이 응용프로그람들을 실행하는 동안에 많은 사용자들이 자기들의 자료를 호출하고 있으면 작업에서 리익을 얻을수 있다. 이 두 동작은 디스크구획들을 분리하기 위한 호출들을 요구한다.
- 어느 구획도 완전한 Linux 를 설치할수 있다는것을 알게 되면 /usr/X11R6(X Windows 나무등록부)과 같은 큰 나무등록부를 명백하게 선택할수 있고 완전한 Linux 체계를 설치할수 있도록 분할된 구획에 그것을 설치할수 있다.
- 많은 새로운 프로그람묶음들이 /opt 에 설치되는 동안 많은 쏘프트웨어들을 설치하면 분할된 구획에 대한 좋은 후보로 되는 /usr 가 빨리 채워 진다는것

을 알수 있다.

- Linux 를 다망한 우편봉사기 혹은 과부하로 쓰이는 다중사용자봉사기로 리용하고 있다면 우편완충대기렬과 체계등록들이 이 등록부에 들어 가고 이러한 형태의 봉사기들이 빨리 증가하므로 /var 그자체의 구획을 제공한다는것을 고려해야한다.

Disk Druid 대신에 fdisk 를 리용

제 4 장에서는 Red Hat Linux 7.1 을 설치할 때 디스크구획을 구성하는데 Disk Druid 를 리용하였다. 어쨌든 Disk Druid 는 Red Hat 설치처리에서만 쓸수 있는 프로그람이다. 보통 Linux 사용자들은 설치처리과정에 그리고 후에 디스크배치를 조정할 필요가 있을 때에 자기들의 디스크구획을 구성하기 위하여 fdisk 를 리용한다.

사실 fdisk 는 Disk Druid 를 대신하는것으로서 Red Hat 가 디스크구획을 구성하는 도구로서 우월하다는것을 인정하는 권위 있는 Linux 사용자들에게는 아주 잘 알려 져 있다.

fdisk 는 아주 복잡하고 강력한 도구(그리고 잘못 쓰면 잠재적으로 위험하고 유해로운 도구)이기는 하지만 구획표를 현시하거나 빈 공간으로부터 새로운 구획을 만들며 구획에 대한 형을 지정하는것과 같은 기본적인 과제처리들을 수행하는 일감은 아주 간단하다.

설치처리과정에 Disk Druid 대신에 fdisk 를 선택하면 첫 화면이 현시되면서 어느 디스크를 가지고 작업하겠는가를 물어 본다. Disk Druid 와는 달리 fdisk 는 한번에 한개의 물리디스크만을 가지고 작업한다. 일단 작업디스크를 선택하면 잘 알려 진 Red Hat 설치프로그람을 잠간 중지하고 그림 5-2 에서 보여 준 초기의 fdisk 화면이 나타나게 된다.

그림 5-2. fdisk 안내화면

주의 이 장의 fdisk 그림들은 이 장에서 론의된것처럼 boot.img 설치플로피디스크로 설치를 시작하는데 따라 달라 진다. 정규적인 도형방식의 Red Hat Linux 설치화면으로부터 fdisk 를 입력하면 표준 Red Hat 설치프로그람의 그림들에서 기대되는것과 정확히 같은 내용들을 보게 된다.

fdisk 는 단순한 지령을 리용하여 조작하는데 매개 지령들은 단일문자이다. 지령을 주려면 간단히 지령을 입력하고 Enter 건을 누른다. 지령이 자기의 과제처리를 실행하기 위하여 추가정보를 요구하면 그것은 빠진 정보에 대하여 재촉한다.

가장 단순한 지령은 m 혹은 ?이다. 어느 문자이든지 fdisk 가 그림 5-3 에서 보여 준 도움말화면을 현시하게 한다. 이 화면에는 이 절에서 서술된 모든 지령을 비롯하여 일반적인 fdisk 의 모든 지령들이 렬거된다.

```
Using /dev/hda as default device!

Command (m for help): m
Command action
     a    toggle a bootable flag
     b    edit bsd disklabel
     c    toggle the dos compatibility flag
     d    delete a partition
     l    list known partition types
     m    print this menu
     n    add a new partition
     p    print the partition table
     q    quit without saving changes
     t    change a partition's system id
     u    change display/entry units
     v    verify the partition table
     w    wirte table to disk and exit
     x    extra functionality (experts only)

Command (m for help):
```

그림 5-3. fdisk 도움말화면

```
Command (m for help): p

Disk /dev/hda: 255 heads, 63 sectors, 525 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot   Begin   Start    End    Blocks   Id  System
/dev/hda1   *        1       1    223   1791216    6  DOS 16-bit >=32M
/dev/hda2          224     224    287    514080
/dev/hda3          288     288    515   1831410   83  Linux native
/dev/hda4          516     516    525     80325   82  Linux swap
/dev/hda5          224     224    287   514048+    6  DOS 16-bit >=32M


Command (m for help):
```

그림 5-4. 구획표

주의

/sbim/fdisk 지령으로 설치한후에 Linux fdisk 편의프로그람을 기동할수 있다. 일부 사용자들은 이미전에 /usr/sbin/cfdisk 편의프로그람을 리용하였다는것을 알고 있다.

구획표의 현시

리용할수 있는 대부분의 fdisk 지령들중의 하나인 p 는 작업하고 있는 능동디스크들에 대한 현행 구획표를 현시한다. 이것은 그림 5-4 의 표와 같다.

현행 디스크에 있는 매개 구획에 대하여 구동기이름, 시작과 끝블로크, 블로크의 구획크기와 체계형들이 현시된다.

빈공간을 리용한 구획추가

하드디스크의 빈 공간을 리용하여 새로운 구획을 추가하기 위하여 n 지령을 리용한다. 그림 5-5 에서 보여 준것처럼 구획형을 재촉하는 재촉문이 나타난다. 일반적으로 Primary 를 선택한다.

```
Command (m for help): n
Command action
    e   extended
    p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (224-525):
```

그림 5-5. 새 구획추가

```
Command (m for help): l


0  Empty             17  Hidden HPFS/NTF  5c  Priam Edisk       a5  BSD/386
1  FAT12             18  AST Windows swa  61  SpeedStor         a6  OpenBSD
2  XENIX root        1b  Hidden  Win95 FA 63  GNU HURD or Sys   a7  NexTSTEP
3  XENIX usr         1c  Hidden  Win95 FA 64  Novell Netware    b7  BSDI  fs
4  FAT16 <32M        1e  Hidden  Win95 FA 65  Novell Netware    b8  BSDI  swap
5  Extended          24  NEC  DOS         70  DiskSecure Mult   c1  DRDOS/sec  (FAT-
6  FAT16             3c  PartitionMagic   75  PC/IX             c4  DRDOS/sec  (FAT-
7  HPFS/NTFS         40  Venix 80286      80  Old Minix         c6  DRDOS/sec  (FAT-
8  AIX               41  PPC PReP Boot    81  Minix / old Lin   c7  Syrinx
9  AIX bootable      42  SFS              82  Linux swap        db  CP/M  /  CTOS  /
a  OS/2 Boot Manag   4d  QNX4 . x         83  Linux             e1  DOS access
b  Win95 FAT32       4e  QNX4 . x 2nd part 84 OS/2 hidden C:    e3  DOS R/O
c  Win95 FAT32 (LB   4f  QNX4 . x 3nd part 85 Linux extended    e4  SpeedStor
e  Win95 FAT16 (LB   50  OnTrack DM       86  NTFS volume set   eb  BeOS  fs
f  Win95 Ext'd  (LB  51  OnTrack DM6 Aux  87  NTFS volume set   f1  SpeedStor
10 OPUS              52  CP/M             8e  Linux LVM         f4  SpeedStor
11 Hidden FAT12      53  OnTrackDM6 Aux   93  Amoeba            f2  DOS secondary
12 Compaq diagnost   54  OnTrackDM6       94  Amoeba BBT        fd  Linux raid auto
13 Hidden FAT16 <3   55  EZ-Drive         9f  BSD/os            fe  LANstep
14 Hidden FAT16      56  Golden bow       a0  IBM Thinkpad hi   ff  BBT


Command (m for help):
```

그림 5-6. ID 구획형을 fdisk 구획형으로 바꾸기

주의 Linux **fdisk** 로는 DOS 의 단일 1 차구획과 대조적으로 4 개의 1 차구획을 만들 수 있다.

일단 구획형이 선택되면 구획번호를 지정하고 마지막으로 시작과 끝디스크실린더를 지정한다. 남아 있는 모든 공간(리용할수 있는 모든 공간이 구동기의 끝에 있다고 가정한다.)을 리용하기 위하여 구획에 대한 처음 및 마지막기정원기둥을 취한다.

구획형의 바꾸기

기정으로 fdisk 에서 작성된 새로운 모든 구획들에는 Linux Native(형번호 83)형이 추가된다. 구획형을 바꾸기 위하여서는 t 지령을 리용한다.

그림 5-6 에서 보여 준것처럼 작업하는 구획에 대하여 수자로 선택할수 있게 재촉되며 그다음에 형 ID 에 대하여 재촉된다. 형 ID 들의 목록을 보려면 그림 5-6 에서 보여 준 목록을 보는 시점에서 L 지령을 리용한다.

가장 일반적으로 리용된 구획형을 표 5-1 에서 보여 주었다.

표 5-1. 일반적으로 리용된 구획형

ID	형
5	확장형
6	DOS 16-bit(MB 보다 더 큰)
7	OS/2 HPFS
8	Windows 95 FAT
c	론리블로크주소(LBA)를 가진 Windows 95 FAT
82	Linux 교환
83	Linux Native

구획의 지우기

때때로 Linux 설치할수 있는 한개이상의 새로운 구획을 만들기 위하여 현존구획을 삭제해 버려야 할 때가 있다. 이를 위해서 간단하게 d 지령을 리용하며 재촉될 때에는 삭제하려는 구획의 번호를 입력한다.

변경위탁

fdisk 로 작업하는 과정에 실제로는 사용자가 만든 어떤 변경도 물리디스크에는 만들어 지지 않는다. 이것은 중요한 자료를 가지고 있는 구획을 부주의로 지워 버렸을

때 자료를 포함한 구획을 완전히 지우기전이라면 이전의 구성에로 되돌아 갈수 있도록 하기 위한 안전한 예방책이다.

그런 리유로 해서 변경은 fdisk 가 탈퇴할 때 어떻게 하겠는가를 명백히 물을 때까지는 실제상 처리되지 않는다. 따라서 fdisk 로부터 탈퇴하기전에 구획표를 주의 깊게 보고 요구대로 정확히 변경되였는가를 확인하여야 한다. 탈퇴하여 변경을 위탁하면 그 변경들은 영구해 지며 실질적으로 초기상태로 복귀할수 없다(리론적으로 fdisk 로 작업하기전에 자기원판에 하드디스크의 구획표를 보관할수 있으며 거기에 보관된것을 가지고 새 구획표를 재배치함으로써 련속적인 오유로부터 회복할수 있다. 즉 어떻든지간에 이것은 복잡한 절차이며 간단하지는 않다. 변경을 위탁하기전에 구획표에 만든 변경들이 정확히 바라는대로인가를 좀 더 확인하여야 한다.).

fdisk 로부터 탈퇴하기 위한 두개의 선택지령은 표 5-2 에서 보여 준다.

표 5-2. fdisk 로부터 탈퇴하기 위한 선택지령

지령	효 과
q	일부 변경들을 처리하거나 보관함이 없이 **fdisk** 로부터 탈퇴한다.
w	모든 변경을 처리하고 보관한 다음 **fdisk** 로부터 탈퇴한다. 이것은 영구적인 작업이기때문에 주의해야 한다.

▶미리 알아두기

결국 실천적으로 Linux 로 작업할수 있는 준비가 되였다. 그 과정의 첫 단계로 X Windows 환경을 보게 된다. X Windows 는 Linux 를 비롯한 Unix 의 모든 조작체계들, 높은 유연성과 마우스로 구동되는 도형사용자대면부(GUI)를 준다.

많은 Linux 책들은 지령행이 실제로 Linux 의 중심이기때문에 그것으로부터 시작한다. 그러나 X Windows 는 새로운 사용자들이 Linux 지령행에 대한 충분한 리해를 가지기전에 Linux 를 빨리 쓸수 있게 한다.

제 6 장 《X Windows 의 개괄》에서는 X Windows 가 무엇을 제공하며 Microsoft Windows 환경과 어떻게 비교되는가를 고찰한다. 다음에 X Windows 의 구성과 리용에로 이행하여 X Windows 의 일반적인 응용프로그람들을 고찰하게 된다.

제 3 편. Linux 에서 탁상환경의 리용

제 6 장. X Windows 의 개괄

지금까지 Linux의 설치에 대해서 보았다면 이제부터는 X Windows환경에 대한 프로쎄스를 보기로 하자. 대다수의 Linux제작자들은 Linux지령행과 Linux구성의 복잡한 세부들을 파악하기전에 X Windows에 직접 들어 가는것은 적당치 않다고 생각할수 있다. 그러나 문서편집, 탁상출판, World Wide Web열람기와 같이 일상적으로 쓰는 프로그람들을 실행할수 있는 생산적인 조작체계로서 Windows나 Mac OS대신에 Linux를 리용할수 있게 해주는것이 바로 X Windows라는것을 알아야 한다.

이 장은 X Windows에 대한 정확한 개념을 주는것으로부터 시작한다. X Windows는 Unix계렬에서 리용하는 도형사용자대면부이다. X Windows는 창문, 대화칸, 단추, 차림표와 같은 일반적인 사용자대면부모델전체를 표시하는 방법을 제공한다. 공학 및 설계응용프로그람을 위한 체계인 Unix워크스테이션에서 도형을 마음대로 만들어 낼수 있게 하는것이 X Windows 이며 PC조작체계시장에서 Linux가 강력한 경쟁자의 지위에 올라 설수 있게 한것도 바로 X Windows이다.

X Windows를 완전히 리해하기 위하여 이 장에서는 Microsoft Windows와 X Windows(X Windows 4의 수정개소들을 포함)의 구체적인 비교를 진행한다. 그다음 X봉사기, 창문관리기, 탁상환경, Motif와 같은 X Windows 의 몇가지 기본구성요소들에 대하여 고찰한다.

X Windows 란 무엇인가

한마디로 말해서 X Widnows는 Unix와 더 나아가서는 Linux의 완전한 도형대면부이다. 그러나 이것이 전부는 아니다. X Windows는 일반사용자들은 물론 X Windows에서 실행되는 쏘프트웨어를 만드는 응용프로그람개발자들에게도 필요한 넓은 범위의 유연한 선택항목을 제공하는 구성가능한 환경이다.

X Windows의 기본개념은 의뢰기-봉사기들이다. 이것은 X Windows가 단일처리기에만 국한되지 않는 환경을 제공한다는것을 의미한다. 응용프로그람들은 망에 있는 각이한 봉사기와 콤퓨터에서 실행될수 있으며 망의 다른 X Windows말단기와 워크스테이션에 현시될수 있다.

응용프로그람의 실행과 그 실행결과표시를 분리시킨것은 응용프로그람의 실행과 표시를 한데 묶어 두고 있는 Windows나 Macintosh에는 없는 개념이다. 이러한 분리의 우점은 망환경에 있으며 이것은 잘 유지되고 강력하면서도 관리하기 쉬운 중심적인 응용프로그람봉사기에서 실행되는 응용프로그람들을 현시하는 복잡한 도형탁상들을 가질수 있게 한다. 사실 바로 이것으로 해서 Unix 및 Linux 그리고 X Windows는 대규모망의 전문적인 체계관리자들속에서 좋은 호평을 받게 되였다.

X Windows환경에서 소개되는 또 다른 개념은 대면부로부터 창문작성을 분리시킨것이다. X Windows체계에서 완전한 GUI를 제공하자면 두가지 응용프로그람들을 실행하여야 한다. 첫번째 응용프로그람은 도형현시장치를 설정(실례로 해상도, 재생속도, 색계조)를 설정하는 X봉사기인데 이것은 창문을 표시하며 마우스움직임과 건반입구를 추적

하고 창문들을 다중화한다. 그러나 X봉사기는 차림표, 창문경계 혹은 이동기구, 창문절환, 최소화나 최대화는 제공하지 않는다. 그림 6-1로부터 창문관리기가 없을 때 X봉사기화면이 어떻게 현시되는가를 알수 있다.

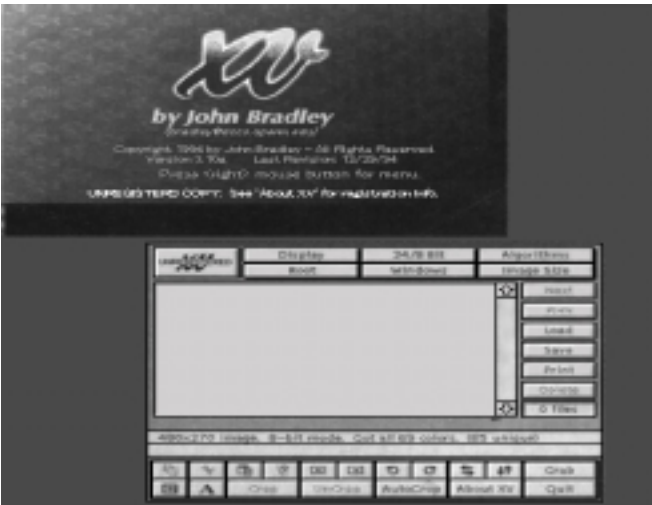



그림 6-1. 창문관리기가 없는 X 봉사기의 현시

그림 6-1은 아주 단순하다. 장식적인 색갈배경도 없고 정교한 창문경계나 차림표조종도 없으며 완전한 GUI를 구성하는 다른 특성들도 없다. 이러한 모든 특성들은 창문관리기라고 하는 다른 응용프로그람이 제공한다. 그림 6-2는 fvwm95창문관리기가 실행되고 있는 완전한 X Windows탁상을 보여 준다. 창문틀, 조종단추 그리고 조종차림표들과 과제띠, 가상탁상체계의 특성들에 주의를 돌려야 한다. 창문관리기는 바로 이러한 모든것들을 제공한다.

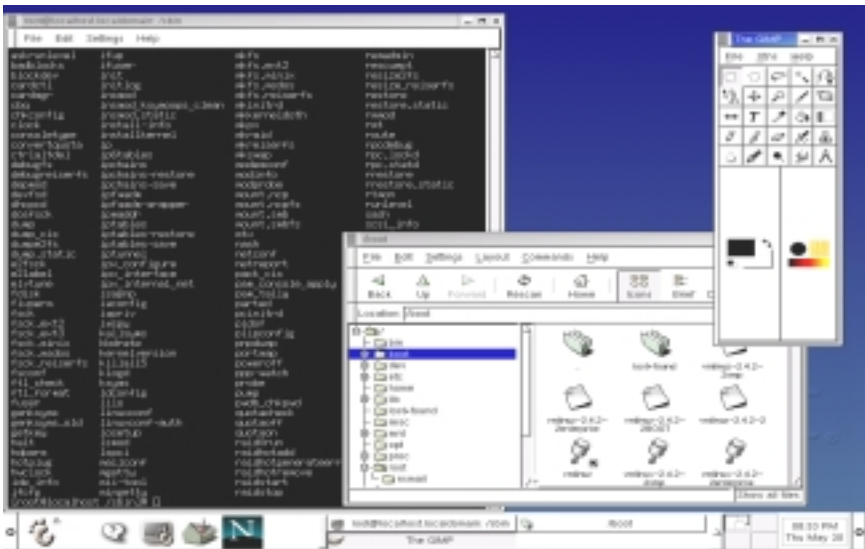



그림 6-2. 창문관리기를 실행하는 X Windows의 현시

창문관리기는 X 봉사기가 X 응용프로그람들과 대화하듯이 표준적으로 미리 정해 진 방법으로 X 봉사기와 대화한다. 이것은 각이한 대면부특성을 가진 각이한 창문관리기들이 이러한 표준적인 방법으로 X 봉사기와 대화할수 있다는것을 의미한다. 마찬가지로 각이한 도형처리기판, 현시장치, 다른 성능특성에 대한 지원을 제공하는 다양한 봉사기들은 응용프로그람들과도 표준적인 방법으로 대화할수 있다.

Microsoft Windows와 X Windows

X Windows에 대한 설명을 통해서 알수 있는바와 같이 Microsoft Windows와 X Windows는 상당히 많은 측면에서 같다고 볼수 있다. 그러나 실제로 이것들사이에는 본질적인 차이가 있다.

실례로 Microsoft Windows는 핵심부로부터 쉘(shell)에 이르기까지 창문환경과 그 밖의 모든것을 포괄하는 완전한 조작체계이다. 그러나 X Windows 는 바로 그 조작체계의 한 부분인 창문환경이다. 또 다른 차이는 대면부에서 찾아 볼수 있다. Microsoft Windows의 대면부가 상당히 엄격한데 비해서 X Windows대면부는 놀라울 정도로 유연하고 전용화가능한 설계양식을 가지고 있다.

류사한 점

그러면 X Windows와 Microsoft의 현재조작체계에서 류사한 점은 무엇인가? 기본류사한 점은 둘다 도형대면부를 갖춘것이고 다중창문으로 작업할수 있는것이다. 이러한 상태에서 사용자는 건반과 문자판이상으로 리용하는 정보로 대화할수 있다. 사용자들은 건반은 물론 마우스를 리용할수 있으며 차림표, 폼, 창문, 대화칸을 결합한 대면부도 만들수 있다.

차이점

두 창문체계사이에는 많은 차이점들이 있다. 기본차이점은 다음과 같다.

- 대면부의 유연성
- 대면부구성에서 세밀조절조종
- 의뢰기-봉사기기술

유연한 대면부

X Windows대면부의 유연성은 사용자들에게 즐거운 환경을 제공한다. 이미 론의된 바와 같이 기본창문층으로부터 사용자대면층을 분리하여 각이한 창문관리기를 작성함으로써 X Windows의 다중대면부를 만들수 있다.

다중대면부가 Microsoft Windows대면부의 전용도구와 같이 겉보기에 미묘한 차이까지 실제로 제공하지는 못하지만 이것이 있으면 창문관리기로부터 창문관리기에로의 사용자대면부를 완전히 재설계할수 있다.

실례로 그림 6-3에 NeXT조작체계의 사용자대면부와 비슷한 창문관리기를 실행하는 창문대면부의 견본을 주었다. 같은 체계에서 실행하고 같은 응용프로그람을 현시하는 그림 6-4와 비교해 보자. 여기서는 일반적인 PC나 Macintosh조작체계와 전혀 다른 사용자대면부가 나타난다.

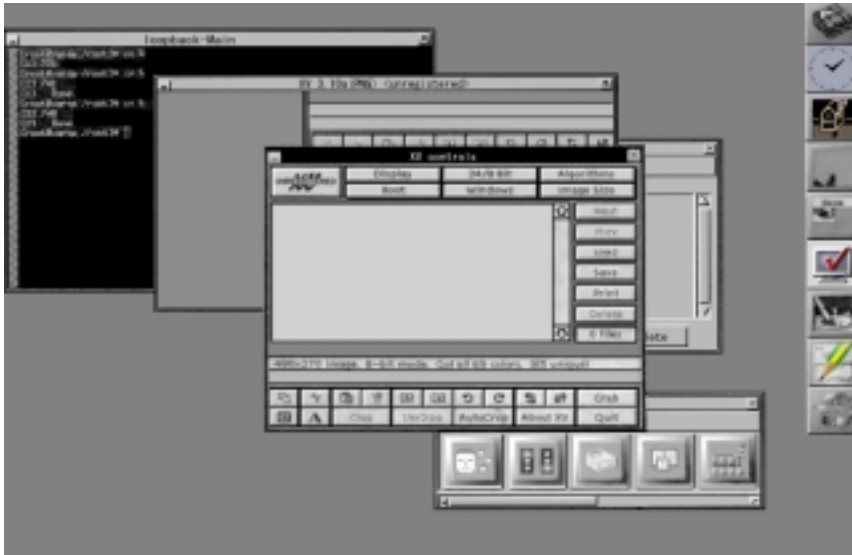



그림 6-3. Afterstep 창문관리기

그림 6-4. Lesstif 창문관리기

이 실례들은 X Windows창문관리기들이 제공한 수많은 시각 및 감각설계의 두가지 형태일뿐이다. 이 장의 마지막에 널리 알려 진 창문관리기들을 간단히 소개한다.

세밀조절조종

X Windows환경의 다른 우점은 창문환경과 대면부의 모든 상황에서 아주 세밀한 조종을 할수 있는것이다. 어떤 설정묶음으로 배경과 전경창문색으로부터 유표색갈, 기정서체, 기정창문크기에 이르기까지의 모든 환경양상을 조종할수 있다. 사용자들은 호상작

용방식들도 정의할수 있다. 실례로 마우스지시자를 움직여 전경이나 배경창문을 자동적으로 절환할수 있다.
그밖에 다른 특성들도 매 응용프로그람이 될수록 가장 편리한 방법으로 실행되도록 여러가지 설정을 만드는 완전한 응용프로그람의 기초우에서 정의할수 있다.
또한 X환경이 기동할 때 어느 창문과 응용프로그람이 기동하며 X Windows가 기동할 때는 어느 창문관리기를 선택하겠는가 하는것도 정의할수 있다.
제7장 《X Windows의 설치와 구성》에서는 기본X Windows의 구성항목들을 고찰한다.

의뢰기-봉사기환경

이미 언급된바와 같이 X Windows는 응용프로그람이 물리적인 현시장치를 구동하는 X봉사기에 대한 의뢰기로 되는 의뢰기-봉사기모델에서 작업한다. 이것은 망환경에 아주 적합하며 망의 임의의 콤퓨터에서 응용프로그람을 실행시키고 그 결과를 다른 콤퓨터에 현시할수 있게 한다.

Microsoft Windows에는 최근까지 이러한 기능이 없었으며 지금은 다만 다중사용자체계인 Windows NT와 2000의 말단봉사기봉사가 제한되고 값이 비싼 류형이 있을뿐이다. Linux, Unix, X세계에서는 가장 낮은 준위의 체계도 X Windows의뢰기-봉사기모델인 의뢰기의 기능을 수행할수도 있고 봉사기의 기능을 수행할수도 있다.

X 봉사기, 창문관리기 및 탁상환경

지금까지 X Windows가 어디에 리용되는가를 알게 되였으므로 이제부터는 X Windows의 기본구성요소들인 X봉사기, 창문관리기, 탁상환경을 고찰해 보자. 이러한 구성요소들과 모듈성이 있으므로 하여 X Windows는 강력하고 유연한것으로 되고 있다.

X 봉사기

이미 본바와 같이 X Windows 체계의 핵심은 X 봉사기이다. X 봉사기는 다음과 같은 여러개의 과제들을 조종한다.

- 다양한 비데오기판과 현시장치의 지원
- X Windows 현시장치의 해상도, 재생속도, 색계조
- 기본창문관리 즉 창문표시와 닫기, 마우스이동과 건반눌림의 추적

이러한 기본성능들을 갖춘 다중 X 봉사기들이 출현하였다. Linux 세계에는 세가지 기본선택물 즉 XFree86, Metro-X, Accelerated-X 가 있다.

XFree86

XFree86 은 표준적인 X 봉사기로서 대부분이 비업무용 Linux 배포물에 속한다. 그러므로 이것은 Linux 에 속해서 무료로 리용할수 있다. 사용자들은 원천코드를 충분히 리용할수 있으며 필요에 따라서는 코드를 변화시킬수 있고 그것을 재배포할수도 있다. Red Hat Linux 7.1 은 Xfree86 4.0.3 을 적재하고 있다. 보조갱신판은 몇달에 한번씩 공개되며 XFree86 Web싸이트의 http://www.xfree86.org로부터 내리적재할수 있다.
XFree86Web 봉사기는 Intel 호환의 x86PC 는 물론 Compaq, Alpha, PowerPC, Sparc, MIPS 처리소편을 가진 콤퓨터에 일반하드웨어의 폭 넓은 지원기능을 제공하도록

설계되였다. Xfree86 의 성능은 그리 우월하지는 않지만 Linux 와 다른 Intel 계 Unix 변종들에서 리용되는 X 봉사기의 기준으로 되고 있으며 그로 하여 X 봉사기는 많은 사람들에게 널리 알려 졌다.

일러두기 이 책을 집필할 당시까지는 MIPS 에 대한 지원기능이 완성되지 못하였지만 XFree86 Project Web 싸이트를 보면 MIPS HOWTO 의 http://oss.sgi.com/mips/mips-howto.html로부터 많은 문서를 리용할수 있다.

XFree86 이 지원하는 일반적인 도형처리기판소편묶음에는 Tseng 회사의 ET 3000, 4000, 6000 과 많은 Trident 소편, Cirrus Logic 준위, Chips 와 Technologies 도형처리소편 등이 있다. 지원하는 장치에 대한 전체 목록은 XFree86 Project Driver 상태페지 http://www.xfree86.org/4.0.3/status.html에서 볼수 있다. 최신판을 리용하려면 4.0.3 으로 바꾸어야 한다.

XFree86 4 X 봉사기가 Linux 를 완전히 자유로운 Unix 와 같은 조작체계로 만들어 나가고 있지만 있지만 사용자들이 회사환경이나 일반단체환경에서 매력을 느끼게 하자면 적어도 몇가지 부족점들을 퇴치해야 한다. XFree86 은 때때로 어려운 구성과 설치를 요구하며 일부 도형처리기판과 현시장치에 대해서는 안정성이나 충분한 지원기능을 담보하지 못하며 가속특성을 가진 도형처리기판의 우점을 살리지 못하는 경우가 있다.

다행히 Red Hat Linux 7.1 은 Xconfigurator 라는 좋은 편의프로그람을 적재하고 있으므로 XFree86 의 구성을 쉽게 할수 있고 일부 장치들을 자동검사하며 그 장치를 위한 가장 좋은 선택항목을 제안한다. 다른 배포물도 이와 비슷한 편의프로그람을 제공한다. 이러한 결함을 가진 장치결합에 이 수법을 쓰지 않으면 작업량은 훨씬 많아 지며 그 경우에 Xfree86 구성을 수동으로 진행해야 한다.

XFree86 4 에서 새로운것은 무엇인가?

XFree86 의 판본 4 는 이전의 판본에 없었던 새로운 특성을 많이 가지고 있다. 이 새로운 특성중에서 일부는 업무용 X 봉사기로부터 따낸것이다. 그 특성은 다음과 같다.

- 한개의 단일 X 봉사기모듈구조설계
- 다중도형처리기판의 지원기능
- 조작체계로부터의 독립
- 3-D 표현
- Free Type 지원기능

한개의 단일 X 봉사기로서의 XFree86 4 는 각이한 형의 도형처리기판들을 재구성하지 않고 리용할수 있게 한다. Metro Link 가 기증한 실행시간적재기를 가진 XFree86 4 는 Linux(혹은 다른 Unix 계의 조작체계)의 장치구동프로그람서고에 의존하지 않는다. GLX 를 확장하면 XFree86 은 Silicon Graphics 회사의 Open GL 3-D 도형처리언어를 통하여 3-D 표현을 지원할수 있다. Free Type 를 지원하는 XFree86 은 이제는 TrueType 서체들을 관리할수 있다.

XFree86 은 Alpha, MIPS, Power PC 처리소편을 리용한 콤퓨터들을 포함하여 보다 많은 가동환경에 대한 지원기능을 확장하고 추가한다. Intel Itanium IA 64-bit 가동환경이 정식 나오면 여기에도 이식될것이다.

주의	이 책에 쓴것과 같이 XFree86 판본 4 는 낡은 XF86Setup 도구를 지원하지 않는다. 대신에 Xconfigurator 와 xf86config 와 같은 편의프로그람이 아주 적절하게 리용된다.

업무용 X 봉사기

Linux 를 완전하고도 자유롭게 해주는데서 기본적인 역할을 하는 XFree86 대상과제 외에도 Linux 세계에는 두개의 업무용 X 봉사기 즉 Metro-X 와 Accelerated-X 가 있다.

이러한 제품들은 여러가지 다양한 도형처리소편 및 기판을 넓은 범위에서 지원할수 있는 우점을 가지고 있으며 이러한 소편과 기판의 가속특성까지도 완전히 지원하고 있다. 또한 Metro-X 와 Accelerated-X 는 보다 지능적인 구성을 제공하며 몇분내에 설치되여 작업할수 있다.

Metro-X Metro Link 에서 나온 Metro-X 는 고성능 X 봉사기의 하나이다. Metro-X 의 우점은 그림 6-5 에서 보여 준바와 같이 도형처리구성대면부가 잘 설계된것, 여러개의 현시장치에 대한 지원기능(표준구성은 4 개의 서로 다른 현시장치를 지원한다.), 접촉식화면에 대한 약간한 지원기능 등이다.

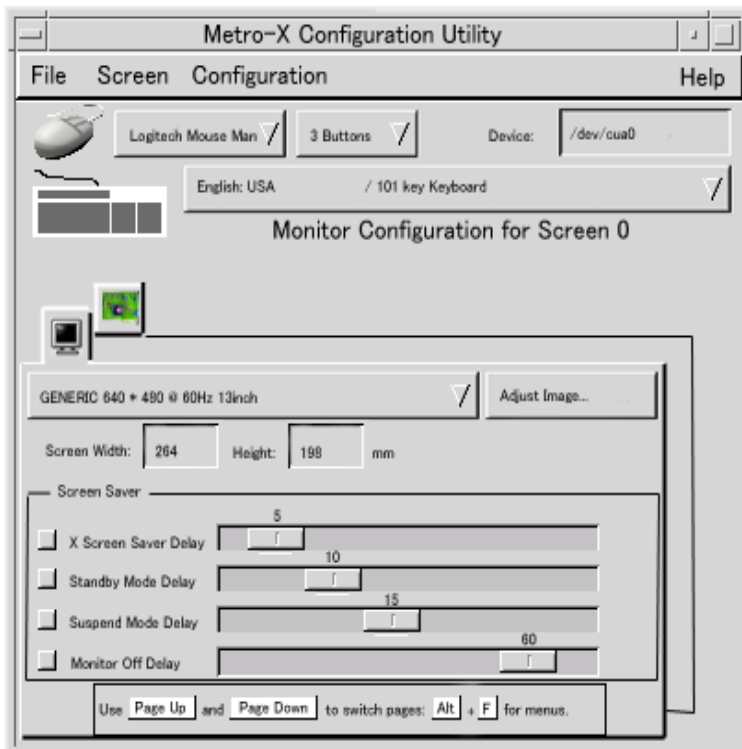



그림 6-5. Metro-X 의 구성창문

지원되는 도형처리기판을 가진 Metro-X 봉사기에 대한 정보는 Metro Link 홈페지 http://www.metrolink.com/.에서 볼수 있다. Linux 에서 리용하는 Metro-X 의 가격

은 39 딸라이다.

Accelerated-X X-Insid 는 Linux 에서 리용할수 있는 X 봉사기를 실제로 제공한다. 여기에는 Linux 를 위한 Accelerated-X 봉사기, 3-D Accelerated-X 봉사기, 여러개의 현시장치를 가진 Accelerated-X 봉사기, 특수한 Laptop Accelerated-X 봉사기가 있다. Laptop Accelerated-X 봉사기는 무릎형콤퓨터에서 볼수 있는 소편묶음가 탁상형콤퓨터의 도형처리기판보다 다양하고 LCD 화면이 탁상형에는 없는 자체의 구동프로그람을 요구하기때문에 의의가 있는것이다.

사실 XFree86 에서는 무릎형콤퓨터현시장치에 대한 지원기능이 완전히 제한되며 LCD 는 약간한 오유로 파괴될수도 있다. Linux 체계를 무릎형콤퓨터에서 리용하려면 Laptop Accelerated-X 봉사기가 지원하는 체계목록에서 요구하는 형을 찾아야 한다.

Accelerated-X 는 60 여개의 회사로부터 400 개이상의 도형처리기판들을 지원하고 있다. 다중현시장치를 가진 Accelerated-X 봉사기는 16 개이상의 화면을 지원하며 매개 화면에 Accelered-X 의 모든 특성을 제공한다.

총체적으로 X-Insid 의 봉사기들의 가격은 경쟁적이다. Accelerated-X 는 99.95 딸라, Laptop 봉사기는 149.95 딸라, 다중현시장치를 가진 봉사기는 199.95 딸라이다. 이에 대한 정보는 X-Insid Web 싸이트의 http://www.xinside.com/.에서 볼수 있다.

창문관리기

창문관리기는 X 봉사기가 제공하지 않은 GUI 의 섬세한 세부까지 제공한다. 여러 특성가운데서 창문관리기는 창문장식(창문의 크기조절, 이동, 닫기, 최소화)과 응용프로그람실행기구(탁상차림표, 조종판, 단추띠와 같은)를 가지고 있다.

이제부터 Linux 에서 리용하는 몇개의 기본창문관리기에 대해서 간단히 요약하여 설명한다. 여기에는 FVWM, fvwm 95, twm, dvwm 등이 있다. Red Hat Linux 7.1 의 표준적인 창문관리기는 Sawfish 이지만 FVWM 과 선택적인 창문관리기도 설치시에 포함한다.

X Windows 의 창문관리기에 대한 개괄은 http://www.plig.org/xwinman/에서 볼수 있다.

FVWM 과 fvwm95

1990 년대 중엽에 들어 서면서 Linux 의 가장 일반적인 창문관리기로는 FVWM 이나 그의 일부 변종이 리용되였다. FVWM 이름앞에 F 가 붙은것에 대해 여러가지 견해가 있다. 어떤 측은 《Freeble》 Virtual Window Manager 를, 다른 측은 《Fine》 Virtual Window Manager 를 의미한다고 하며 또 다른 측은 F 의 의미가 오래동안 론의되지 않았으므로 중요하지 않다고 본다.

FVWM 은 Motif 창문관리기(Motif Window Manager 즉 MWM 은 업무용이다.)와 비슷하게 설계된 유연하고 전용화가 가능한 창문환경을 제공하는 경량급창문관리기이다. FVWM 은 창문관리기를 확장한 모듈체계와 여러개의 가상탁상을 제공하며 최근의 판본 2.2.5 는 실행중의 구성변경과 창문특성전용화를 제공한다. FVWM 의 홈페지는 http://www.fvwm.org/이다. 그림 6-6 에 FVWM 탁상의 한가지 실례를 주었다.

그림 6-6. FVWM 탁상의 실례

FVWM 은 거의 모든 Linux 배포물에 있으며 시감각을 가진 새로운 창문관리기의 기초로 널리 쓰이고 있다. 그의 변종은 다음과 같다.

fvwm 95 Windows 95 와 같은 시감각으로 설계된다.

Afterstep NeXT 환경과 같은 시감각으로 설계된다.

SCWM(Scheme Configurable Window Manager) Scheme 언어를 리용하여 구성할수 있다.

그림 6-7 에 기동중에 있는 fvwm95 의 화면을 주었다.

그림 6-7. Fvwm 95 는 Windows 95 와 같은 환경을 제공한다.

Fvwm95 는 FVWM 의 판본 2 에 기초하고 있으며 유연하면서도 구성이 쉽다. 그것은 FVWM 의 모듈을 계속 지원하며 과제띠와 같은 Windows 95 형의 특성을 얻는데 필요한 모듈도 추가하고 있다. Fvwm95 에 대한 정보는 http://www.plig.org/xwinman/fvwm95.html 에서 볼수 있다.

주의	FVWM2 프로그람묶음는 Red Hat Linux 7.1 설치의 두번째 CD-ROM 에서 찾을수 있다(이 책에는 포함되지 않았다.).

twm

Tab Window Manager(twm)는 종종 초기개발자의 이름을 따서 Tom 의 Window Manager 라고 부르기도 한다. 이 창문관리기는 대용창문관리기로서 Red Hat Linux 7.1 에 속하는 기능환경이다. 그림 6-8 에 전형적인 twm 탁상을 주었다.

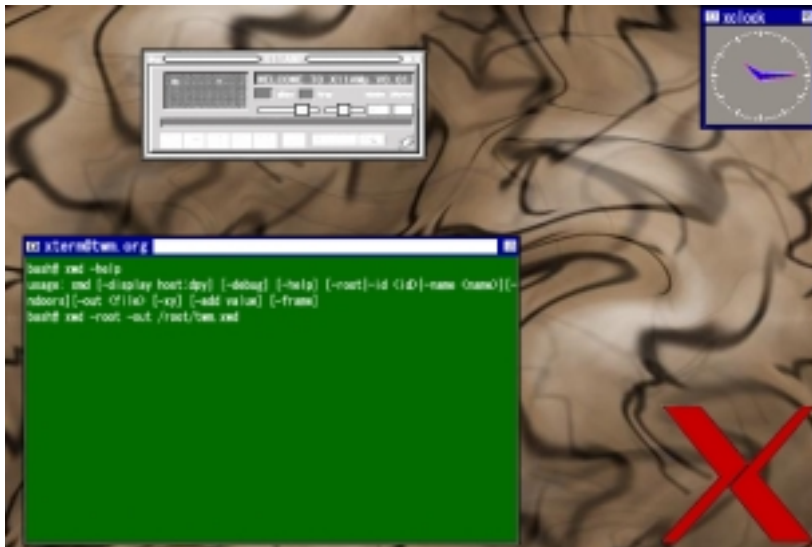



그림 6-8. twm 탁상

twm 창문관리기는 누름형 혹은 점구동형건반초점 그리고 사용자정의건과 단추결합을 비롯하여 사용자의 X Windows 창문관리기에서 기대하는 대부분의 주요특성을 제공한다. 그러나 대면부가 시각적으로 단순해 보이고 어떤 측면에서는 제한성이 있는 느낌을 준다. Vtwm 은 가상탁상용을 포함하는 twm 의 판본이며 ftp://ftp.x.org/R5contrib/vtwm-5.3.tar.gz에서 내리적재하여 리용할수 있다.

olvwm

OpenLook Virtual Window Manager 인 olvwm 은 몇년간 Sun Systems 회사에서 표준창문관리기로 리용하던 Open Look Window Manager(olvwm)의 확장형이다. Sun systems 가 지금은 Motif 와 Common Desktop Environment(이 장의 마감에서 론의된다.)에 치중하고 있지만 OpenLook 대면부가 독특한것으로 하여 여전히 많은 사용자들속에서 리용되고 있다. olvwm 은 OpenLook 프로그람에 가상탁상환경의 지원기능을 추가하고 있다. 그림 6-9 에 olvwm 의 실례를 보여 주었다.

olvwm 의 대면부는 특히 사용자가 마우스단추로 차림표와 창문에 응답하는데서는 불편하다. 이것이 OpenLook 가 Sun 을 제외한 기타 세계에서 광범한 인기를 얻지 못한 한가지 리유이다. olvwm 은 ftp://ftp.x.org/R5contrib/olvwm4.tar.z에서 내리적재할수 있다.

그림 6-9. olvwm 의 대면부

AfterStep

Afterstep 는 FVWM 코드의 다른 변종이다. 이 제품은 이전의 창문관리기인 Bowman 에 기초하고 있으며 NeXT 가동환경에 NeXTSTEP 창문관리기의 감각을 제공하기 위하여 설계된것이다. 그림 6-10 에 Afterstep 탁상의 실례를 주었다.

NeXTSTEP 의 주요특성은 제목띠, 단추, 경계틀, 차림표의 형태, NeXTSTEP 와 같은 아이콘 및 단추띠 등이다. 이것은 FVWM 판본 1 의 코드에 기초하고 있으므로 FVWM 판본에 있는 모듈은 Afterstep 에서도 리용할수 있다. pvwm 95 와 olvwm 과 같은 창문관리기와는 달리 Afterstep 대한 개발연구는 오늘도 계속 진행되고 있다.

Afterstep 홈페지는 http://www.afterstep.org/에 있다.

AmiWm

이전의 Amiga 콤퓨터사용자들과 그 대면부를 좋아하는 사용자들은 AmiWm 창문관리기를 리용할수 있다. 그림 6-11 에서 볼수 있는것처럼 AmiWm 은 Amiga 작업환경을 모방한것이다. Amiga 에서 볼수 있는것과 같은 다중화면을 지원한것으로 하여 AmiWm 이 있으면 Aimga 사용자들은 X Windows 를 쉽게 리용할수 있다.

http://www.lysator.liu.se/~marcus/amiwm.html에 있는 AmiWm 홈페지로부터 AmiWm 에 대한 정보를 조사할수 있고 쏘프트웨어를 내리적재할수 있다.

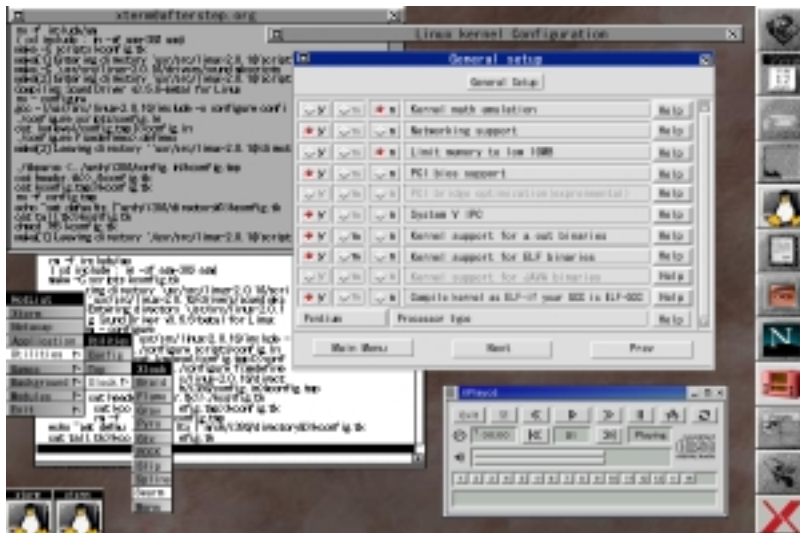



그림 6-10. Afterstep 탁상

그림 6-11. AmiWm 창문관리기

Enlightenment

Enlightenment 는 관습을 릉가하여 창문관리기의 개발을 시도하는 웅대한 계획이다. 이것이 출현함으로써 사용자는 보기에 매력 있는 환경을 제공 받고 창문관리기의 기능으로부터 모양까지 모든것을 정의할수 있게 되였다. 그림 6-12 와 6-13 에 Enlightenment 환경의 두가지 가능한 구성을 주었다. Enlightenment 를 방조하는 많은 특성에는 사용자들이 수동으로 시감각을 만들수 있는 능력과 새로운 특성을 쉽게 삽입할수 있는 능력이 있다.

Enlightenment 가 거대한 대상과제이고 시각적으로 황홀한 대면부를 제공하지만 사

용자가 종종 새로운 서고들을 콤파일하고 설치할것을 요구하기때문에 미숙한 사용자에게는 그 설치가 좀 어려울수 있다. Enlightenment 홈페지는 http://www.enlightenment.org/에 있다. 선행단계(이 책을 집필할 당시의 판본은 0.16 이다.)에서 저자들이 Enlightenment 를 리용하였으므로 오유와 있을수 있는 충격들을 경고하고 있다. 그러나 여전히 이것은 흥미 있는 대상과제이며 창문관리기의 독특한 개념이다.

그림 6-12. Enlightenment 창문관리기

그림 6-13. 사용자는 Enlightenment 환경을 완전히 전용화할수 있다.

Sawfish

Sawfish 창문관리기는 GNOME 탁상환경을 동반하는 기정적인 새로운 창문관리기이다. 이 경우에 Enlightenment 와 같이 Sawfish 도 LISP 프로그람언어를 리용하는 도구

로 매개 창문을 구성하는 능력을 최대화한다. GNOME 탁상의 부분으로서 Sawfish 조종은 그림 6-14 에 보여 준바와 같이 GNOME 조종쎈터와 통합된다. 제 10 장 《고급한 GNOME 구성》에서 GNOME 조종쎈터와 Sawfish 를 구체적으로 설명한다.

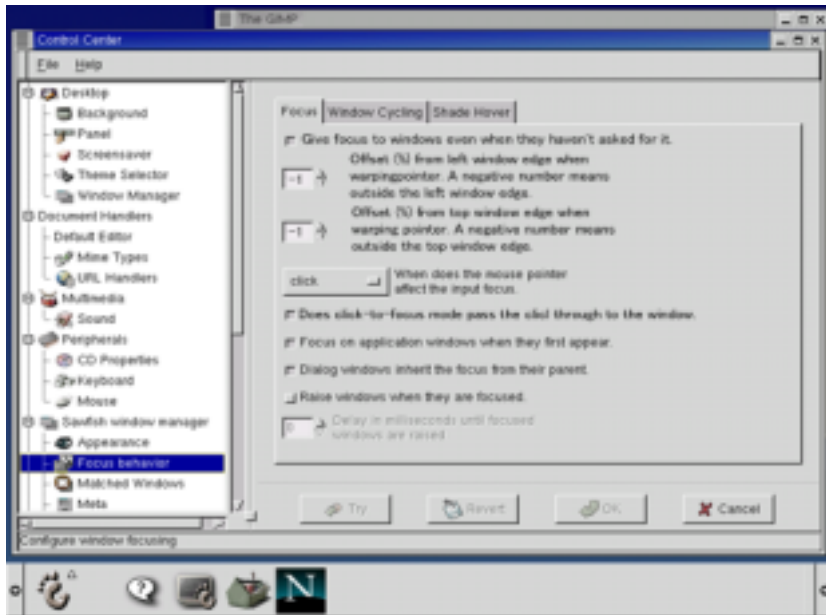



그림 6-14. 사용자들은 Sawfish 환경을 완전히 전용화할수 있다.

Enlightenment 와 마찬가지로 Sawfish 도 역시 기동과정에 작업할수 있다. 현재판본은 0.38 이다. 더 구체적인 정보를 보려면 http://www. sawfish.org에 있는 Sawfish 홈페지로 가야 한다.

탁상환경

탁상환경은 창문관리기우에 놓인다. 탁상환경의 목적은 완전히 밀착된 GUI 를 제공하는것이다. 간단한 창문관리기에서는 다른 조작체계에서 볼수 있는 응용프로그람과 특성호상간에 대한 시감각의 일관성을 담보하지 못하는데 이 특성은 응용프로그람호상간 끌어다놓기와 교차응용프로그람자료의 채워넣기와 같다.

Linux 의 다양한 환경은 안전하고 통합된 GUI 를 제공하고 공통적인 시감각으로 쏘프트웨어를 제작하는 응용프로그람개발자들에게 목적하는 가동환경을 제공하며 끌어다놓기와 같은 통합특성을 포괄하고 있다.

두가지 종류의 탁상환경 즉 K Desktop Environment 와 GNOME 은 무료로 리용할수 있는데 이것은 사실상 표준 Linux 탁상환경의 경쟁자로 출현하였다.

K Desktop Environment

K Desktop Environment(KDE)는 Common Desktop Environment(이 부문의 마감에서 론의된다.)에 대한 무료선택을 제공한다. 결과 KDE 는 X 응용프로그람과 일치한 시감각을 주는 창문관리기와 함께 파일관리기, 말단모방기, 현시장치체계와 같은 응

용프로그람을 결합한다. 그림 6-15에 전형적인 KDE 탁상을 주었다.

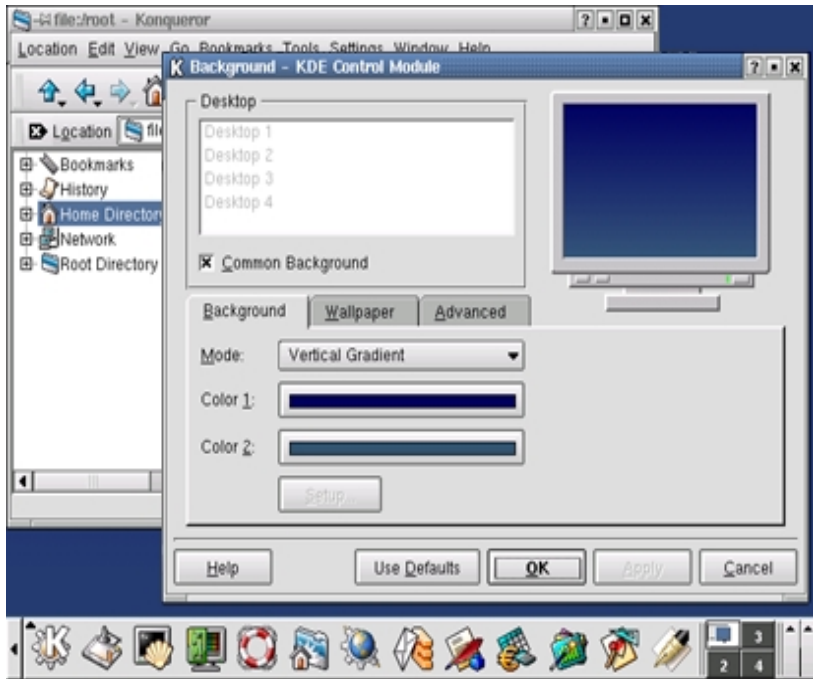



그림 6-15. KDE 탁상

KDE는 Caldera, S.u.S.E., Corel을 비롯한 많은 Linux배포물에 대한 기정탁상환경이다. 그것은 대부분의 다른 배포물에 대한 대리탁상환경이기도 하다.

KDE가 후에 계획하는 항목은 표준화와 개발자들이 특성은 물론 모양까지도 일치한 응용프로그람을 만들수 있는 개발환경이다. KDE는 매력적이고 전문적인 환경을 제공하며 요구하는 응용프로그람이 KDE용으로 개발되면 그것은 업무탁상용으로 된다. KDE Web싸이트는 http://www.kde.org/이다. 제 11장 《KDE의 리용》에서는 Red Hat Linux 7.1의 KDE에 대하여 상세하게 설명한다.

GNOME(The Network Object Model Environment)

GNOME은 Linux의 포괄적이며 무료적인 탁상환경을 개발하기 위한 노력의 결과물이다. 창문관리기가 있는 KDE와는 달리 GNOME은 창문관리기와 독립이다. 이것은 창문관리기개발자들이 자기의 창문관리기에 GNOME을 완전히 지원할수 있도록 프로그람대면부를 제공한다.

Red Hat Linux 7.1은 Sawfish 창문관리기를 리용하는 기정탁상환경으로서 GNOME을 적재한다. 다른 창문관리기와 KDE도 이 배포물에 있지만 기정설치가 GNOME과 Enlightenment를 리용하기때문에 이 책에서도 목적하는 탁상으로서 GNOME과 Enlightenment를 리용한다. GNOME 리용에 대한 소개는 제 8장 《GNOME과 X Windows의 리용》에서 보게 된다.

MWM

MWM(Motif Window Manager)은 Motif 배포물에 속하는 업무용창문관리기이다. Motif 의 완전한 환경은 이 장의 다음절에서 상세하게 고찰한다.

Common Desktop Environment

Common Desktop Environment(CDE)는 AIX, Compaq unix, HP/UX, Solaris 를 포괄하는 다양한 Unix 가동환경에서 도형처리환경과 개발무대의 표준화를 위한 거대한 대상과제이다. CDE 는 현재 Linux 용으로서 X-Inside (http://www.xinside.com/)로부터 Desktop Graphical Interface 를 리용할수 있다. 가격이 49 딸라인 이 업무용응용프로그람은 X-Inside 의 Accelerated-X 봉사기를 요구한다.

Motif 에 일치하는 도형처리환경외에도 CDE 는 관리순서를 표준화하기 위한 밀착된 도구들과 응용프로그람들을 제공하며 사용자가 도형처리작업환경을 쉽게 구성하고 관리할수 있게 한다. CDE 가 X 환경에 가져다 준 강화된 기능들에는 끌어다놓기성능과 다른 GUI 조작체계들에서 볼수 있는 등록부와 아이콘형태가 있다.

Motif 란 무엇인가

Linux 체계에 X Windows 응용프로그람들을 설치하기 위해 Web 을 찾아 보느라면 불가피하게 Motif 서고나 Motif 도구묶음(toolkit)을 리용하는 응용프로그람을 보게 된다.

Motif 는 X Windows 응용프로그람들에 대한 일관한 정책을 제공하기 위하여 1980 년대 말에 열린 쏘프트웨어협회(OSF :Open Software Foundation)가 소개한 X Windows 의 개발환경이다. Motif 는 개발자들이 자기의 응용프로그람을 개발하는데 리용하는 도구묶음제품(Widget)을 제공한다. 이러한 제품서고를 리용하여 Motif 개발자들은 일치한 시감각으로 Motif 정책에 충실한 응용프로그람들을 만들어 낼수 있었다.

최근에 Motif 는 《공개(open)》사용허가밑에서 재공개되였는데 이것은 Motif 서고들에 기초한 응용프로그람들을 만들고 배포하려는 개발자들이 그 특전을 보상할 필요가 더는 없다는것을 의미한다. 그러나 새로운 Open Motif 뒤에 있는 조직인 Open Grap 은 여전히 이러한 언어와 서고들의 리면에서 문서를 팔고 있다. 그리고 많은 Linux 개발자들은 Linux 와 같은 무료조작체계를 위한 쏘프트웨어를 개발하는데 Motif 특허를 더는 쓸수 없다는것을 판단하였다.

주의	Open Graup 은 쏘프트웨어를 표준화하기 위한 산업기구이다. 그것은 Open Source 사용허가의 후원지와 련관이 없다. 실제로 그들은 Motif 사용허가가 Open Source 사용허가를 만족시키지 못한다는것을 인정한다.

도구묶음과 서고들외에도 Motif 배포물에는 자체의 창문관리기 MWM 이 있는데 그 후에 FVWM 시감각이 표준으로 되였다. 그림 6-16 은 MWM 탁상의 실례를 보여 준다.

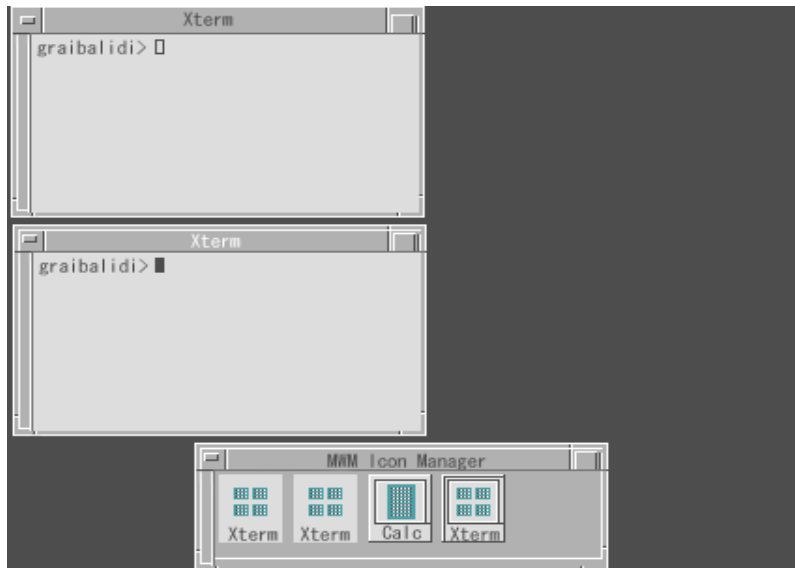



그림 6-16. MWM 탁상

Motif 가 필요한가

대다수의 Linux 사용자들에게는 아마 Motif 를 리용할 필요가 없다. Motif 는 다음과 같은 경우의 사용자들이 리용한다.

- Motif 도구묶음을 리용하여 응용프로그람을 개발하려고 하는 개발자들
- Motif 서고가 있기를 바라면서 응용프로그람을 실행하려고 하는 사용자들 Motif 서고를 리용하는 대부분의 업무용응용프로그람에는 필요한 Motif 코드를 그 응용프로그람에 삽입(사용자가 자체의 사용허가를 필요로 하지 않기때문이다.)하든지 아니면 실행시간(runtime)Motif 서고들의 완전한 판본이 있어야 한다.

http://www.opengroup.org/에 있는 Open Group Web 싸이트로부터 Open Motif 를 내리적재할수 있다. Metron Link 는 현재 Motif 도구들의 각이한 판본을 가진 프로그람묶음을 99 딸라이상으로 팔고 있다.

Motif 의 선택

Linux 세계에서 대개가 그러하듯이 Linux 개발자와 사용자들이 Motif 를 사지 않고 무료로 리용할수 있는 선택을 가지도록 노력해야 한다. Lesstif Project 는 Motif 판본 1.2 와 모순되지 않고 GNU 일반공개사용허가밑에서 리용할수 있는 제품을 개발하였다.

Lesstif 는 아직 충분한 완성품으로는 되지 못하였지만 (이 책을 집필할 때의 판본은 0.92.26 이다) 이미 일부응용프로그람들을 개발하고 몇개의 쏘프트웨어를 실행하는데 사용될수 있었다. Lesstif 홈페지(http://www.lesstif.org/)우에서 Lesstif FAQ 는 Lesstif 가 여전히 완전하지 못하다는것을 민감하게 지적한다. 그렇다 하더라도 Motif 를

리용한 일부응용프로그람들은 Mosaic 2.7 과 GNUCash 개인재정관리기를 포괄하는 Lesstif 0.92.26 으로 작업할수 있다.

주의 Motif 를 무료로 내리적재할수 있다. 그러면 《Motif 가 무료로 리용되고 있는 지금 왜 Lesstif 를 리용하는가?》 무료쏘프트웨어협회는 새로운 Modif 사용허가가 대부분의 현재 Linux 배포물에서 리용이 금지되여 있고 따라서 Motif 는 실제로 무료가 아니라고 믿고 있다. 그러므로 여전히 Motif 대신에 Lesstif 를 요구하고 있다.

▶미리 알아두기

X Windows 에 대한 이 론의는 Linux 조작체계를 리용하기 위한 출발점으로 된다.

다음의 두개 장에서는 Red Hat Linux 7.1 의 표준탁상환경 즉 GNOME 을 리용하는 X Windows 대면부의 기초를 파악하게 된다. 사용자들은 여기에서 X Windows 를 설치하는 방법을 배우게 되고 몇가지 가장 일반적인 X Windows 와 GNOME 응용프로그람들에 대한 실험을 진행하게 된다.

제 7 장 《X Windows 의 설치와 구성》에서는 단계별로 X Windows 의 설치와 구성에 대한 안내를 준다.

제 7 장. X Windows 의 설치와 구성

X Windows 에 대한 개념을 가진 현재의 상태에서 X Windows 가 될수록 빨리 실행되는 체계를 가지도록 쏘프트웨어를 설치하고 구성할 때가 되였다.

이 장에서는 Linux 의 배포물로에서 적재된 X Windows 를 얻는 방법을 고찰한다. 다음에 X Windows 가 콤퓨터에서 원만하게 작업하도록 XFree86 을 구성하는 방법을 설명한다.

X Windows 의 얻기

이 책과 함께 제공된 Red Hat Linux 7.1 을 리용하여 제 4 장 《 Red Hat Linux 7.1 의 설치》에서 제안된 주요구성요소들을 모두 설치하였다면 이 부분을 읽을 필요가 없으며 《X Windows 를 구성하기 위한 준비》절에로 넘어 갈수 있다.

Red Hat Linux 7.1 CD-ROM 에 의한 설치

Linux 를 설치할 때 Red Hat Linux 7.1 을 사용하고 있지만 X Windows 가 설치되여 있지 않으면 X Windows 의 설치를 매우 간단히 할수 있다. 여기서 개괄된 순서는 CD-ROM 구동기를 호출하는 제 14 장 《파일다루기》에 서술된 기술에 따른다.

rpm 의 리용

이 절에서는 Red Hat Linux 와 Red Hat 형프로그람묶음을 관리하기 위해서 다른 배포물을 적재하는 도구인 **rpm**(Red Hat Package Manager)을 리용하게 된다.
rpm 을 리용하면 새 프로그람묶음을 설치하고 설치되여 있는 프로그람묶음들을 해제하며 거기에 무슨 파일이 들어 있는가 하는 프로그람묶음에 대한 정보를 볼수 있다.
프로그람묶음을 설치하기 위해서 다음의 지령을 리용한다.

```
$ rpm -i 프로그람묶음-파일-이름
```

설치된 프로그람묶음의 목록을 보기 위해서 다음의 지령을 리용한다.

```
$ rpm -qa | more
```

이 지령은 단번에 한개 화면에 설치된 프로그람묶음들의 목록을 표시한다. 또한 공백건을 눌러서 다음화면으로 넘어 갈수 있다. 목록에서 보게 되는 프로그람묶음이름은 설치할 때 리용된 파일이름이 아니다. 그렇지만 그것들은 다음의 지령으로 프로그람묶음의 설치를 해제하는데 리용된다.

```
$ rpm -e 프로그람묶음-파일-이름
```

Red Hat Linux 7.1 CD-ROM 으로 X Windows 를 설치하기 위한 첫번째 단계는 Linux 가 그것을 호출할수 있게 CD-ROM 구동기를 설치하는것이다. 이 처리는 구동기(IDE/ATAPI, SCSI 혹은 전용)의 형태와 구동기가 설치되는 방식에 따라 변화될수 있기때문에 보는것처럼 간단하지 않다. CD-ROM 선택항목들에 대한 더 구체적인 론의는 제 19 장 《Linux 다매체》에서 보게 된다.

오늘 다매체 PC 에서는 IDE CD-ROM 구동기를 쓰는것이 일반적이기때문에 IDE CD-ROM 구동기를 가지고 있다고 볼수 있다. 오늘날 판매되고 있는 대부분의 탁상형콤퓨터에는 한개의 하드디스크만이 있기때문에 CD-ROM 구동기가 제 2 의 부속장치로 설치된다고 가정할수 있다.

이러한 가정하에서 CD-ROM 구동기를 /dev/hdb 장치로 볼수 있다. 기타 구동기형과 구성에 따라 자기의 구동기가 무슨 장치인가를 결정하는것은 제 19 장에서 론의된다.

구동기에 Red Hat Linux 7.1 CD-ROM 을 넣고 디스크를 설치하고 Linux 에서 호출하기 위하여 다음의 지령을 준다.

```
$ mount /dev/hdb /mnt/cdrom
```

CD-ROM 의 내용을 보고 호출가능성을 확인하기 위해서 다음의 지령을 준다.

```
$ ls /mnt/cdrom
```

결과는 다음과 같이 나타난다.

```
autorun dosutils     RedHat               RELEASE-NOTES.es RELEASE-NOTES.ja
boot.cat images      RELEASE-NOTES        RELEASE-NOTES.fr PRM-GPG-KEY
COPYING README       RELEASE-NOTES.de     RELEASE-NOTES.it  TRANS.TBL
```

주의

CD-ROM 장치가 무엇인지 알수 없으면 보통 다음과 같은 지령을 준다.
mount /dev/cdrom /mnt/cdrom
다음지령을 실행하면 CD-ROM 이 실제적인 장치에 련결되였다는것을 확인할 수 있다.
ls –l /dev/cdrom

다음단계는 Red Hat Linux 7.1 을 구성하는 모든 프로그람묶음들이 들어 있는 CD-ROM 의 등록부를 변경시키는것이다. 이 등록부는 RedHat/RPMS 보조등록부이다. 다음의 지령을 리용한다.

$ cd /mnt/cdrom /RedHat/RPMS

다음지령을 실행하면 긴 파일목록이 나타나는데 여기의 모든 파일의 끝에는 RED HAT Package Manager 파일이라는것을 가리키는 확장자 .rpm 이 있다.

$ ls

이제는 X Windows 프로그람묶음들을 설치할 준비가 되였다고 볼수 있다. 다음은 설치할 프로그람묶음들이 어느것인가를 지정해야 한다. 일반적인 XFree86 파일은 XFree86-4.0.3-5.i386.rpm 프로그람묶음에 들어 있다. 또한 모든 X Windows 서체는 다음과 같은 파일에 있다.

```
XFree86-100dpi-fonts-4.0.3-5.i386.rpm
XFree86-75dpi-fonts-4.0.3-5.i386.rpm
```

이외에도 공유된 XFree86 서고와 X Windows 서체봉사기들은 다음파일에 있어야 한다.

```
XFree86-libs-4.0.3-5.i386.rpm
XFree86-xfs-4.0.3-5.i386.rpm
```

XFree86 구성편의프로그람은 Xconfigurator-4.9.27-1.i386.rpm 파일에, X Windows 프로그람묶음은 XFree86-tools-4.0.3-5.i386.rpm 에 있다.

XFree86 4 는 더이상 X 봉사기프로그람을 분리시키지 않는다. 그러나 현행 판본까지도 3.3x 판본이 지원하는 모든 현시장치와 도형처리기판을 지원하지 못하고 있다. 자기가 가지고 있는 XFree86 프로그람묶음의 공개기록들을 다시 조사해 보아야 한다. 4.0.3 판본의 공개기록은 http://www.xfree86.org/4.0.3/RELNOTES.html 에서 볼수 있다. XFree86 4 프로그람묶음에 대한 각이한 판본이 있으면 4.0.3 으로 바꾸어야 한다.

주의

XFree86 4 는 도형처리기판지정형 X 봉사기들에 대한 프로그람묶음을 더는 분리시키지 않는다.

지금까지 설치하려는 파일목록을 만들었으므로 이제는 이 파일들을 설치하기 위하여 rpm-i 지령을 리용할수 있다.

```
$ rpm -i  XFree86-4.0.3-5.i386.rpm
$ rpm -i  XFree86-100dpi-fonts-4.0 3-5.i386.rpm
$ rpm -i  XFree86-75dpi-fonts-4.0 3-5.i386.rpm
$ rpm -i  XFree86-1ibs-4.0.3-5.i386.rpm
$ rpm -i  XFree86-xfs-4.0.3-5.i386.rpm
```

```
$ rpm -i  Xconfigurator-4.9.27-l.i386.rpm
$ rpm -i  XFree86-tools-4.0.3-5.i386.rpm
```

일단 이 지령이 실행되면 필요한 모든 X Windows 파일이 체계에 설치되여야 한다.

주의 하나의 선택은 XFree86 RPM 파일들을 인터네트로부터 내리적재하는것이다. RPM 들을 위한 포괄적인 원천은 http://www.rpmfind.net 에 있는 rpm 《탐색엔진》이다. 이것의 결함은 새로운 Linux 응용프로그람의 공개와 그의 RPM 형식의 공개사이에 보통 지연이 존재하는것이다. 다음부분에서는 XFree86 의 최신판본을 어떻게 내리적재하고 설치하는가를 보여 준다.

인터네트에 의한 설치

Linux 배포물형식보다 더 새로운 판본의 XFree86 을 설치하려면 필요한 파일들을 인터네트에서 내리적재하여 설치할수 있다. XFree86 판본 4.0.3 의 설치는 두 단계로 나누어 진행한다. 먼저 4.0.2 판본을 설치하고 다음에 《갱신》으로서 4.0.3 판본을 설치할수 있다. XFree86 의 더 선진적인 판본을 설치하고 리용할 때도 과정은 이와 류사하다. XFree86 은 XFree86 대상과제에 의하여 개발되며 http://www.xfree86.org Web 싸이트에서 이에 대한 정보를 볼수 있다.

주의 제 22 장 《Linux 를 인터네트에 접속》에서는 인터네트와의 접속을 고찰하며 제 23 장 《World Wide Web 의 리용》에서는 Netscape 의 리용을 고찰한다. Linux 에 아직 망환경이 준비되여 있지 않지만 그 콤퓨터에 Windows 나 DOS 가 설치되여 있으면 Windows 나 DOS 의 인터네트접속을 리용하여 파일을 내리적재한 다음 Linux 로부터 Windows 나 DOS 구획을 호출하여 그 파일들에 대한 설치를 시작할수 있다.

그림 7-1. XFree86 의 Web 페지

그림 7-1 은 XFree86 의 Web 페지를 보여 준다. http://ftp-stud.fht-esslingen.de/pud/Mirrors/ftp.xfree86.org/XFree86/에 있는 이 Web 싸이트를 비롯한 수많은

싸이트들로부터 XFree86 의 최신판본에 대한 파일이 있는 등록부를 호출할수 있다. 여기서 다음과 같은것이 들어 있는 수많은 Linux 판본서고를 볼수 있다.

Linux-alpha-glibc21 Compaq 의 Alpha 처리기를 위한 Linux.

Darwin Apple 이 PowerPC 와 Intel 형콤퓨터에 제공하는 Darwin 조작체계를 위한 Linux.

Linux-ix86-libc5 일반적으로 더 오래전의 Linux 배포물에서 Intelx86 처리기를 위한 Linux.

Linux-ix86-glibc2x 더 새로운 Glibc 서고를 리용하는 Intelx86 처리기를 위한 Linux 판본. 대부분의 보다 새로운 배포물은 이 GNU C 언어서고들을 리용한다. Red Hat 7.1 은 Glibc 2.2 서고를 리용한다.

XFree86 판본 4.0.2 의 설치

첫 단계로 기본프로그람묶음을 설치하는것이다. 이 경우에 프로그람묶음의 판본은 4.0.2 이다. 내리적재싸이트로부터 판본 4.0.2 등록부를 호출한다. 다음 자기의 Linux 판본에 적용할 서고를 선택한다. 의문되는 점이 있으면 Xinstall.sh 스크립트를 내리적재하고 sh Xinstall.sh -check 지령을 실행시켜 볼수 있다. 이 지령에 의해 자기의 배포물에 리용되는 (내리적재하려는) 서고를 알수 있다. 일단 자기의 판본을 위한 서고가 선택되면 다음의 파일들이 내리적재된다.

```
Xinstall.sh   extract      Xetc.tgz
Xbin.tgz      Xdoc.tgz     Xfnts.tgz
Xlib.tgz      Xman.tgz     Xfenc.tgz
Xvar.tgz      Xxservtgz    Xmod.tgz
```

거의 모든 경우에 자기의 콤퓨터에 XFree86 을 설치하려면 Xinstall.sh 스크립트를 리용한다. 내리적재한 파일이 있는 등록부(/dos/tmp)에 가서 다음지령으로 스크립트를 실행시킨다.

```
$ cd /dos/tmp
$ sh Xinstall.sh
```

우의 지령이 실행되지 않으면 다른 지령들을 리용하여 XFree86 을 설치할수도 있다. 먼저 설치에 필요한 등록부들을 만든다. 이를 위해서 mkdir 지령을 준다. 지령에 대한 상세한 내용은 제 14 장에서 본다.

```
$ mkdir /usr/X11R6
$ mkdir /etc/X11
```

다음 내리적재한 파일에 있는 설치프로그람을 실행시킨다. Windows 구획의 tmp 등록부에 내리적재된 파일이 있고 이미 제 4 장에서 /dos 로 구획이 설치되였다고 본다. 그러면 X Windows 등록부로 가기 위해 다음지령을 실행시킬수 있다.

```
$ cd /user/X11R6
```

다음단계는 내리적재한 압축형태의 파일로부터 파일을 뽑아 내는것이다. 다음지령으로 이 작업을 진행할수 있다(이 지령중의 일부는 이 책의 마감에 론의되는것들이며 지금은 X Windows 를 실행하기 위한 준비단계이므로 론의를 피한다.).

```
$ chmod +x /dos/tmp/extract

$ /dos/tmp/extract -C /usr/X11R6 X[a-df-uw-z]*.tgz
$ /dos/tmp/extract -C /etc/XII Xetc.tgz
$ /dos/tmp/extract -C /var Xvar.tgz$
$ ln -s /etc/XII/app-defaults /usr/XIIR6/1ib/XII
$ ln -s /etc/XII/fs /usr/XIIR6/1ib/XII
$ ln -s /etc/XII/lbxproxy /usr/XIIR6/1ib/XII
$ ln -s /etc/XII/proxymngr /usr/X11R6/1ib/XII
$ ln -s /etc/XII/rstart /usr/X11R6/1ib/XII
$ ln -s /etc/XII/twm /usr/X11R6/lib/XII
$ ln -s /etc/XII/xdm /usr/XIIR6/1ib/XII
$ ln -s /etc/XII/xinit /usr/XIIR6/1ib/XII
$ ln -s /etc/XII/xsm /usr/XIIR6/lib/XII
$ ln -s /etc/XII/xserver /usr/X11R6/1ib/XII
$ /sbin/ldconfig /usr/X11R6/1ib
$ /usr/XIIR6/bin/mkfontdir /usr/X11R6/lib/XII/fonts/misc
```

지금은 이 모든 지령들이 신비스러워 보이지만 이 실행의 결과 X Windows의 필요한 파일이 Linux체계에 설치된다.

XFree86 판본 4.0.3의 설치

두번째 단계에서는 판본 4.0.2의 갱신판본인 XFree86 판본 4.0.3을 설치하는 방법을 보기로 한다. 이미 설명한 내리적재싸이트의 판본 4.0.3 등록부를 찾아 가 보자. 그리고 이미 설명한 같은 서고항목을 리용하여 다음의 파일을 내리적재한다.

```
Xinstall.sh        extract          Xupdate.tgz
Xdocupd.tgz        Xdrivers.tgz
```

거의 모든 경우에 콤퓨터에 XFree86 을 설치하기 위하여 판본 4.0.3 에서 내리적재한 설치프로그람스크립트 Xinstall.sh 를 리용할수 있다. 앞에서와 마찬가지로 내리적재된 파일이 들어 있는 등록부(/dos/tmp)에 옮겨 가서 스크립트를 실행시킨다.

```
$ cd /dos/tmp
$ sh Xinstall.sh
```

이것은 XFree86 판본 4.0.2 가 이미 설치되였을 때에만 가능하다. 이상의 설치가 완성된후에 Xconfigurator 나 xf86config 를 리용하여 X Windows 를 구성한다. 이 두가지 편의프로그람은 다음절에서 고찰한다.

X Windows를 구성하기 위한 준비

실제적으로 X Windows 를 구성하려면 X Windows 를 최적화하는 처리과정에 현시장치의 안전을 담보하기 위하여 하드웨어에 대한 몇가지 중요한 정보를 수집해야 한다. 필요한 정보는 다음과 같다.

- 비데오기판의 제작과 모델
- 현시장치의 제작과 모델
- 비데오기판의 비데오기억기
- 마우스형태
- 현시장치의 수평이동범위
- 현시장치의 수직이동범위

이러한 정보가 없이 XFree86 을 구성하면 잠재적인 파괴를 가져 올수 있는 가능성이 있다. 이 정보를 하나씩 보자.

비데오기판의 제작과 모델

이러한 성능지표들은 그다지 어렵지 않게 얻을수 있다. 일반적으로 기판설명서에서 이러한 정보를 찾을수 있다.

상표이름이 없는 비데오기판을 구하였다면 제작이 맞지 않는다. 이 경우에는 기판에서 어떤 소편을 리용하고 있는가를 조사하기 위하여 비데오기판의 문서를 찾아 본다. 보통 리용되는 소편들로서는 Cirrus Logic, S3, Chips, Technologies 등을 들수 있으며 이밖에 다른것들도 있다. 모든 경우에 소편의 모델번호와 소편제작회사가 기록되여야 한다.

주기판에 비데오를 지원하는 콤퓨터라면 소편의 제작회사와 모델번호를 알기 위하여 그 콤퓨터나 주기판문서를 조사해 보아야 한다.

문서가 없는 경우에는 비데오하드웨어를 탐색하는 XFree86 편의프로그람인 SuperProbe 를 리용할수 있다. 이 지령을 실행하면 자기의 하드웨어와 관련된 X 봉사기, 소편, RAMDAC 를 알수 있다. 프로그람을 실행시키기 위하여 다음의 지령을 준다.

```
$ SuperProbe
```

현시장치의 제작과 모델

주어 진 현시장치의 제작과 모델에 대한 설명서를 검색한다. 자기의 현시장치가 구성쏘프트웨어가 조사할수 있는 형이라면 이 정보는 XFree86 을 구성하는데 리용할수 있다.

비데오기판의 비데오기억기

기판의 기억용량은 비데오기판설명서에서 볼수 있다. 현대콤퓨터들에서는 일반적으로 용량이 4MB 이상이며 일부 PC 에서는 8MB 혹은 32MB 이다. 낡은 콤퓨터나 비데오기판에는 256kB 만큼 작은것도 있다.

설명서에서 기판의 기억용량을 볼수 없다면 PC 의 전원을 투입하고 다시 기동해야 한다. 콤퓨터에 전원을 투입하면 첫 화면에 제일 먼저 나타나는것이 기판의 제작, 모델번호, 비데오기억기용량이다.

리용하려는 마우스의 형태

이 정보는 마우스가 X Windows 에서 동작하도록 하는데 필요한것이다. 제 4 장에서 Linux 를 설치할 때 마우스를 이미 구성했다면 X Windows 를 구성하는데도 같은

정보가 있어야 한다.

마우스를 구성하지 못했다면 마우스의 형태를 검사해 보아야 한다. 있을수 있는 마우스의 형태에는 다음과 같은것들이 있다.

- Microsoft 호환직렬마우스(보통 두단추형)
- MouseSystem 호환직렬마우스(보통 세단추형)
- 모선(bus)마우스
- PS/2 마우스
- Logitech 직렬마우스
- Logitech MouthMan(Microsoft 호환)
- Microsoft IntelliMouse

거의 모든 마우스들은 이러한 형태의 마우스와 호환할수 있다.

주의	이 책에서는 보는바와 같이 Xconfigurator 도 아니고 xf86config 도 아닌 USB 마우스를 구성할수 있다. Red Hat Linux 7.1 에는 이러한 목적에 리용할수 있는 /usr/sbin/mouseconfig 편의프로그람이 있다.

주의	굴개가 달린 마우스가 있는 경우에는 일부 우편함설치구성을 해야 한다. 어떤 경우에는 굴개가 달린 가운데단추처럼 설정되는것도 있다. /etc/x11/XF86Config 나 /etc/X11/XFConfig-4 파일에서 ZAxisMapping 변수가 있는 행의 첫 머리에 붙은 #기호를 없애야 마우스굴개로 창문을 흐르게 할수 있다. 그외의 더 상세한 정보는 http://www.xfree86. org/4.0.3/mouse.html 을 참고해야 한다.

현시장치의 수평동기범위

이 정보는 주어 진 현시장치를 구성쏘프트웨어가 명백히 지원하지 못하는 경우에 특별히 중요하다. 이 값을 모르고 (다음절에서 설명하는 수직동기범위도 마찬가지이다.) 현시장치에 신호를 보내기 위하여 비데오기판을 조정하면 현시장치가 파괴될수 있다.

사용설명서의 현시장치의 특성부분이 아니면 수평동기범위(때때로 수평주사범위라고도 함)에 대한 정보는 비데오방식을 설명한 절에서 찾아 볼수 있다. 이 정보는 30~～70kHz 와 같은 kHz 단위의 값범위에서 지정한다. 이 범위는 대체로 30kHz 근방이다. 이 값은 최대가능한 값으로 설정할수도 있고 최대보다 약간 작은 값으로 설정할수도 있다.

현시장치의 수직동기범위

수평동기범위와 같이 수직동기범위(때때로 수직주사범위라고도 함)도 현시장치의 위험을 막는데서 본질적이다. 사용설명서의 수평동기범위와 같은 범주에서 정보를 찾을수 있다. 이 값은 50～160Hz 와 같은 Hz 범위(수평동기와 같이 KHz 값으로 표시되지 않는

다.)로 표시된다(일반적으로 비교적 큰 값은 극히 낡은 현시장치가 아니면 75Hz 이상으로 될것이다. 72Hz 이상의 수직주사속도를 보장하지 못하는 현시장치는 아물거리는 현상때문에 짧은 시간동안 화면을 들여다 봐도 눈피로감이나 머리아픔이 올수 있다.). 반드시 수직동기범위의 최대가능한 값으로 설정해야 좋다.

Xconfigurator에 의한 XFree86 구성

Red Hat Linux 에는 Xconfigurator 라는 구성프로그람이 있는데 이것을 리용하면 XFree86 보다 훨씬 더 쉽게 설치할수 있다. 먼저 Xconfigurator 를 리용하여 X Windows 를 구성하고 다음에 Xconfigurator 를 가지지 못한 체계에서 XFree86 을 구성해 보자.

Xconfigurator 는 XFree86 을 구성하는 처리를 아주 단순하게 해주는 차림표구동도구이다. 그 프로그람을 실행하기 위하여 다음의 지령을 주면 그림 7-2 에 보여 준 첫 화면이 나타난다.

```
$ Xconfigurator
```

그림 7-2. Xconfigurator 의 첫 화면

주의	X Windows 구성의 모든 단계들을 성과적으로 완성하려면 뿌리사용자가 되여야 한다.

이 프로그람을 리용하기 시작하려면 첫 화면에서 Enter 건을 눌러야 한다. Xconfigurator 는 먼저 도형처리기판을 검색한다. 이것이 성공하면 그림 7-3 과 같은 화면이 나타난다.

화면에 나온 정보가 무엇인지 알면 도형처리기판과 관련된 X 봉사기를 가를 필요가 없다. 다른 한편 첫 차림표를 그림 7-4 에 주었다.

그림 7-3. PCI 도형처리기판의 조사결과

Xconfigurator 4.0.27 - (C) 1997-2001 Red Hat, Inc. and others
Choose a Card
Pick a Card from the list below (Or choose "Unlisted Card"
at the bottom of the list if your card isn't listed)
0 the Max MAXColor S3 Trio64V+ S3 Trio64+
2-the-Max MAXColor 6000 ET6000
3DLabs Oxygen GMX PERMEDIA 2
3DVision-i740 AGP Intel 740
3Dlabs Permedia2 (generic) PERMEDIA 2
928Movie S3 928
ABIT G740 8MB SDRAM Intel 740
AGP 2D/3D V. 1N, AGP-740D Intel 740
AGX (generic) AGX-014/15/15
Ok Back
<Tab>/<Alt-Tab> between elemets | <Space> selects | <F12>nextscreen

그림 7-4. 비데오기판차림표

여기서는 그 체계가 장비하고 있는 비데오기판이나 소편을 선택한다. 영어자모순서로 표시된 비데오기판과 소편들의 목록이 많은 경우에는 웃방향건이나 아래방향건(혹은 PgUp 이나 PgDn 건)을 리용하여 화면을 흐르게 한다.

자기의 기판이나 소편을 정확히 맞추는것이 중요하다. 정확하지 않은 류사한것으로 맞추어 놓으면 실패할수 있다. 실제로 《류사한》기판은 자기의것과 명백히 차이날수 있다. 이 경우에는 목록의 중간부분에 있는 Generic VGA Compatible 이나 목록의 마지막에 있는 Unlisted Card 를 선택할수 있다. 일단 선택되면 다음화면으로 넘어 가기 위하여 Enter 를 누른다.

Unlisted Card 를 선택한 경우 그림 7-5 에서 보여 준것과 같은 두번째 차림표화면이 나타나며 여기에서 자기의 기판에 적합한 봉사기를 선택할수 있다. 어느 봉사기를 선택하겠는지 확신이 없는 경우 앞에서 본 X Windows 설치절에서 이미 선택된 일반봉사

기를 리용한다. 대부분의 현대적인 다매체콤퓨터들은 SVGA 봉사기를 리용한다.

그림 7-5. X 봉사기차림표

일단 기판과 봉사기를 선택하였으면 다음단계에서는 그림 7-6 에서 보여 준 현시장치선택차림표화면에서 자기의 현시장치를 지정한다.

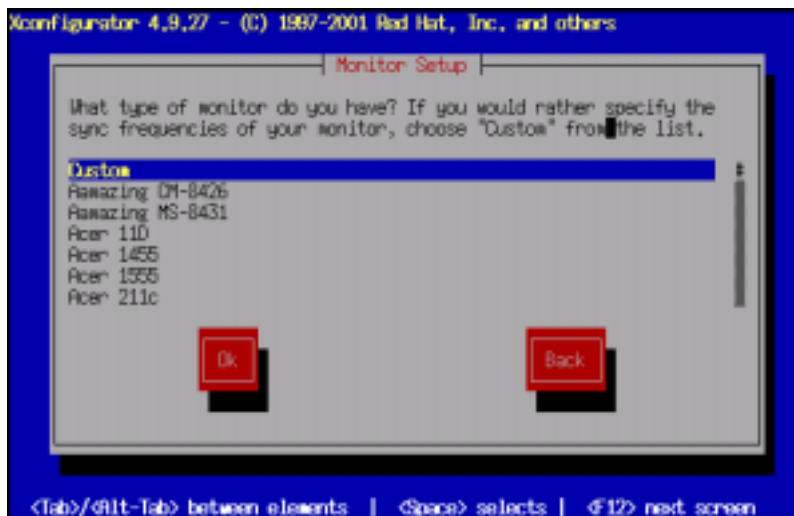



그림 7-6. 현시장치선택차림표

이 목록에서 자기의 현시장치를 찾을수 없는 경우 다음의 세가지 방법으로 선택할수 있다.

- Custom 을 선택하고 자기체계의 수평 및 수직동기범위를 규정한다(이미 설명된 바와 같이).
- 자기의 현시장치에 대한 최대해상도나 동기범위의 일반지식이 있으면 일반적인 현시장치가운데서 하나를 선택할수 있다. 각이한 일반현시장치들은 기본해상도

(640×480 으로부터 1600×1200)를 리용한다. 대부분의 무릎형콤퓨터들에 장비된 LCD 현시장치에는 다른 일반현시장치선택항목들이 리용된다(주해를 보시오.).

경고 LCD 영상표시장치나 무릎형콤퓨터를 구성할 때 특별히 주의해야 할것은 최대해상도와 동기범위를 변경시키는것이다. 잘못 설정하면 LCD 화면이 항시적으로 위험할수 있다. LCD 나 무릎형콤퓨터를 적재한 문서를 참고하거나 최대해상도나 동기범위를 설정하기전에 제작자에게 문의해야 한다.

- 자기의 현시장치에서 쓸수 있는 동기범위와 다중동기방식현시장치를 모르는 경우 가장 낮은 준위의 수직동기속도를 가진 Generic LCD Panel 이나 Generic Standard VGA 를 선택해야 한다(이것들은 가장 낮은 공통적인 특징이며 마지막선택이다.).

선택이 결정되면 자기의 비데오기판에 대한 정확한 해상도와 색계조를 조사할수 있다. 현대의 PCI 기판은 이것을 지원하며 이런 기판인 경우에는 조사해 볼 여지가 있다. 낡은 기판이 있거나 조사가 실패한 경우 수동으로 적합한 설정을 구성할수도 있다.

현시장치선택차림표에서 Custom 을 선택한 경우와 조사를 선택하지 못하였거나 조사가 실패한 경우 현시장치가 피해를 받을수 있다는것을 상기시키는 주의화면이 나타난다. 그림 7-7 에서와 같은 수평동기범위차림표를 처리하기 위하여 Enter 를 누른다.

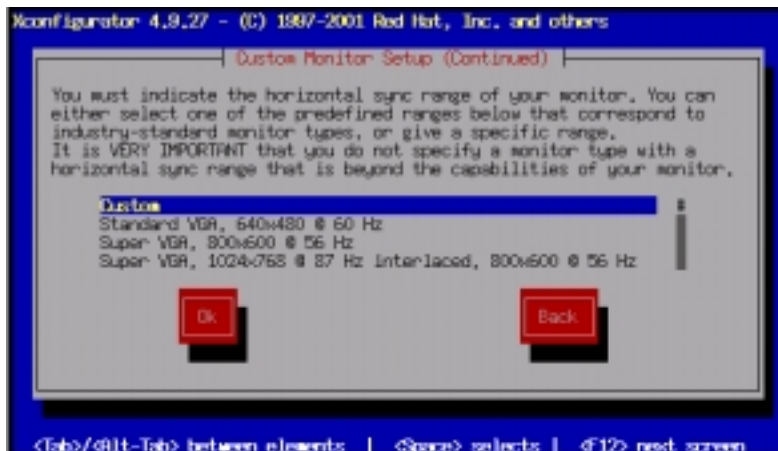



그림 7-7. 수평동기범위차림표

목록을 보고 정확히 자기의 현시장치에 맞는 항목을 선택한다. 선택하였으면 Enter 를 누른다. Xconfigurator 에서의 선택은 범위값만이 아니라 최대가능한 값으로도 지정된다. 이 정보는 자기가 가지고 있는 현시장치에 대한 사용설명서의 동기범위와 같은 위치에서 볼수 있다. 가장 높은 해상도와 가능한 동기속도항목을 선택하는것이 중요하다. 현시장치의 사용설명서에 없는 값을 지정하는 항목은 결코 선택하지 말아야 한다.

한편 현시장치의 사용설명서에 동기범위가 지정되여 있으면 Custom 을 선택한다. 다음차림표에서 정확한 수평 및 수직동기범위들을 지정할수 있다. 일단 선택되면 그림 7-8 과 같은 수직동기범위의 차림표가 나타난다.

그림 7-8. 수직동기범위차림표

이 차림표로부터 가장 큰 값이 현시장치에 대한 수직동기범위의 가장 큰 값을 초과하지 않고 가장 작은 값이 현시장치의 수직동기범위의 가장 작은 값보다 더 작은 값을 선택한다.

Xconfigurator 가 처리가 시작될 때 기판을 검출했으면 Xconfigurator 가 제공하는 다음화면에서 자기의 기판이 가지고 있는 가능한 비데오방식을 조사할것인가 안할것인가를 선택한다. 일반적으로 낡은 기판들을 가지고 이것을 시도하거나 최근의 기판에 문제가 있는 경우라도 대부분 새로운 기판들은 성과적으로 조사된다.

여기서 가장 좋은 방법은 조사해 보는것이다. 오유가 있으면 통보문을 내보내며 여기서 묘사된것처럼 수동방법으로 구성해야 한다. 조사가 성공하면 여러단계들을 보존한다.

조사를 선택하지 못하거나 조사가 실패하면 그림 7-9 에서 보여 준것과 같은 비데오기억기차림표를 현시하는 화면이 나타난다. 여기에서 구성준비절에서 본것처럼 자기의 비데오기판에 맞는 값을 선택해야 한다.

그림 7-9. 비데오기억기차림표

다음화면은 박자소편설정을 위한 선택을 제공한다. 자기의 기판에 박자소편이 있는지 그리고 박자소편의 정확한 형이 무엇인지 잘 알수 없는 경우 그림 7-10 에서와 같이 기정 No Clockchip Setting 을 설정하는것이 좋다.

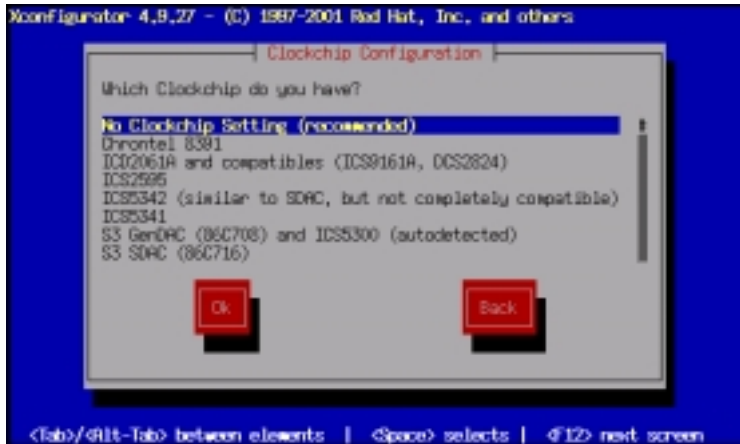



그림 7-10. 박자소편차림표

처리시작시에 Xconfigurator 가 기판을 발견하지 못하면 그것이 X -probeonly 지령을 실행하는지 알아 보아야 한다. 이 지령은 박자소편을 리용하는것이 불가능할 때 필요한 박자정보를 검출한다. 그러나 모든 기판들이 성과적으로 조사될수 없다. 조사를 지원하지 못하는 기판을 가진 경우 조사는 실패하게 된다. 하드웨어를 조사하는 과정에 실패통보문이나 Xconfigurator 가 파괴되면 기판에 대한 조사를 더 할수 없다.

박자소편을 선택하면 그림 7-11 과 같은 화면이 나타나는데 여기에서 적합한 비데오방식을 선택할수 있다.

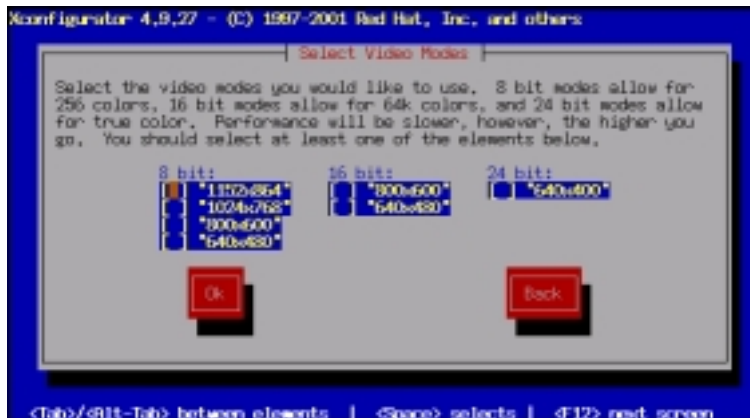



그림 7-11. 비데오방식화면

더 높은 비트계조를 선택하면 더 많은 색갈이 표시될수 있지만 비데오성능은 오히려 더 낮아 진다. 마찬가지로 해상도를 더 높게 설정하면 탁상우에 더 많은 자원을 표시할수 있지만 문자나 아이콘의 크기가 작아 진다. 대체로 자기체계에 대해서 가장 좋은 설

정을 요구하는데 일반적으로 그것은 14in 현시장치에서는 640×480, 15in 에서는 800×600, 17in 에서는 1024×768 이다. 충분한 도형처리기판기억기가 있으면 Netscape 나 The GIMP 와 같은 색집약응용프로그람에 대해서 16bit 색이나 24bit 색이 아주 좋다.

선택하려는 비트계조사이는 Tab 건으로 이동하고 해상도사이는 방향건으로 이동한다. 방식을 선택하려면 공백건을 리용해야 한다. 방식을 선택하고 Tab 건을 리용하여 OK 단추로 이동한 다음 Enter 건을 누른다.

주의	한개이상의 비데오방식을 선택하면 X Windows 에서 방식들사이 절환은 Ctrl+Alt+더하기건과 Ctrl+Alt+덜기건으로 할수 있다.

마지막으로 Xconfigurator 는 구성을 시험한다. 몇초후에 화면이 번쩍한다. 구성이 성공하면 다음의 재촉문이 화면에 나타난다.

```
Can you see this message? Automatic timeout in 10 seconds.
```

통보문을 볼수 있다면 Yes 를 찰칵한다. 다음 Xconfigurator 는 재기동할 때 X 봉사기를 시작하는 선택항목을 준다. 재기동되면 Linux 는 도형처리접속화면을 준다.

이것은 구성처리를 완성했다는것을 의미한다. 구성파일이 작성되였다고 통보하면 Enter 를 눌러서 Xconfigurator 를 끝낸다. Red Hat Linux 7.1 에서 작성된 파일은 XF86Config 라고 하며 /etc/X11 등록부에 놓인다.

XFree86 을 Xf86config 로 구성

Xconfigurator 는 XFree86 을 구성하기 위한 한가지 방도만을 제공하며 이 방법은 Red Hat Linux 에서만 리용할수 있다. Xfree86 그자체는 구성프로그람과 함께 제공되는데 이것은 Xfree86 즉 xf86config 가 들어 있는 Linux 배포물에서 리용할수 있다.

Xf86config 도 차림표구동형이지만 Xconfigurator 보다 리용하기가 좀더 불편하다. 그러나 xf86config 가 있으면 마우스와 건반을 구성할수 있다. Xf86config 를 실행하려면 다음의 지령을 주어야 한다.

$ **xf86config**

그러면 그림 7-12 에 준것과 같은 환영화면이 나타난다.

```
This program will create a basic XF86Config file, based on menu selections
you
make.

The XF86Config file usually resides in /usr/X11R6/etc/X11 or /etc/X11, A
sampleXF86Config file is supplied with Xfree86; it is configured for a
standard VGA card and monitor with 640*480 resolution. This program will ask
for a pathname when it is ready to write the file.

You can either take the sample XF86Config as a base and edit it for your
configuration, or let this program produce a base XF86Config file for your
configuration and fine-tune it.

Before continuing with this program, make sure you know what video card
you have, and preferably also the chipset it uses and the amount of video
memory on your video card. SuperProbe may be able to help with this.

Press enter to continue, or ctrl-c to abort.
```

그림 7-12. xf86config 첫 화면

일러두기 xf86config 를 리용하여 X Windows 를 구성할 때 모든 단계들을 성공적으로 완성하려면 뿌리사용자가 되여야 한다.

주의 xf86config 는 구성처리과정에 화면에 밀집된 본문을 현시한다. 첫 화면이 바로 그의 첫 실레이다. 따라서 xf86config 와 친숙하지 못한 경우에도 한 화면에서 대부분의 정보를 읽을수 있으므로 그 중요성을 잘 알수 있다.

첫번째 차림표는 그림 7-13 에서 보여 준 마우스규약차림표이다. 적당한 선택항목을 요구하는 번호로 입구하고 Enter 를 누르면 마우스와 련관된 질문들이 화면에 나타난다. 즉

- 《Do you want to enable ChordMiddle?》Logitech 에서 만든 세단추 Microsoft 호환마우스가 있으면 가운데단추로 작업할수 있도록 ChordMiddle 을 선택해야 한다. 그렇지 않으면 왼쪽과 오른쪽단추만을 리용할수 있다.
- 《Do you want to enable Emulate3Buttons?》 Microsoft 호환규약과 같은 두단추마우스와 두단추마우스규약을 리용하는 경우 이 항목을 리용할수 있다. 이것은 중간단추를 모의한다. 중간단추는 오른쪽과 왼쪽마우스단추들을 함께 눌러서 리용하며 일부 X Windows 응용프로그람에서 리용될수 있다.

```
Hirst specify a mouse protocol type. Choose one from the following list;
1. Microsoft compatible (2-button protocol)
2. Mouse Systems (3-button protocol)
3. Bus Mouse
4. PS/2 Mouse
5. Logitech Mouse (serial, old type, Logitech protocol)
6. Logitech MouseMan (Microsoft compatible)
7. MM Series
8. MM HitTablet
9. Microsoft IntelliMouse
If you have a two-button mouse, it is most likely of type 1, and if you have a
three-button mouse, it can probably support both protocol 1 and 2. There are
two main varieties of the latter type: mice with a switch to select the protocol,
and mice that default to 1 and require a button to be held at boot-time to select
protocol 2. Some mice can be convinced to do 2 by sending a special sequence
to the serial port (see the ClearDTR/ClearRTS options).
Enter a protocol number:
```

그림 7-13. 마우스규약차림표

- 마우스와 관련된 마지막질문은 마우스장치에 대한 요구이다. Red Hat Linux 를 설치할 때 리용했던 마우스와 같은것을 리용하고 제 4 장에서 설치할 때 마우스를 이미 구성했다면 **/dev/mouse** 를 입력할수 있다. 이것은 기정이므로 여기에서는 그대로 Enter 를 누를수 있다.

주의 Red Hat Linux 7.1 에 들어 있는 xf86config 의 판본은 USB 장치를 인식하지 못한다. USB 마우스와 Red Hat Linux 7.1 이 있으면 /usr/sbin/mouseconfig 편의프로그람을 리용해야 한다.

다음의 많은 절에서 건반구성을 고찰한다. 첫 화면은 그림 7-14 에서 보는바와 같이 가능한 건반형에 대한 차림표이다.

```
Please select one of the following keyboard types that is the better
description of your keyboard. If nothing really matches,
choose 1 (Generic 101-key PC)

1 Generic 101-key PC
2 Generic 102-key (Intl) PC
3 Generic 104-key PC
4 Generic 105-key (Intl) PC
5 Dell 101-key PC
6 Everex STEPnote
7 keytronic FlexPro
8 Microsoft Natural
9 Northgate Omnikey 101
10 Winbook Model XP5
11 Japaness 106-key
12 PC-98xx Series
13 Brazilian ABNT2
14 HP Internet
15 Logitech iTouch
16 Logitech Cordless Desktop Pro
17 Compaq Internet
18 Microsoft Natural Pro

Enter a number to choose the keyboard.
```

그림 7-14. 건반차림표

자기의 형에 맞는 건반을 선택한다. 일반 PC 용 101 건반이나 104 건반은 북아메리카 영어를 리용하는 대부분의 콤퓨터들에 가장 적합한 후보자로 된다. 다음차림표는 각이한 언어나 지역어들을 보여 준다. 나라별 목록은 한 화면을 넘기때문에 Enter 건을 리용하여 다음화면을 볼수 있다.

주의 그러나 영어건반이 아닌 건반과 비표준건반은 제 12 장 《고급한 X Windows 구성》에서 보게 되는 수동방법으로 구성해야 한다.

이제는 건반을 얼마든지 구성할수 있다. 《Do you want to select additional XKB options?》라는 재촉문에 y 를 입력하면 다국어건반에 대응하는 각이한 건반그룹을 설정할수 있다.

건반구성이 끝나면 현시장치의 수평 및 수직동기범위를 지정해야 한다. 제시된 첫번째 차림표는 그림 7-15 에서 준 수평동기범위차림표이다.

```
You must indicate the horizontal sync range of your monitor. You can either
Select one of the predefined ranges below that correspond to industry-
Standard monitor types, or give a specific range.

It is VERY IMPORTANT that you do not specify a monitor type with a horizontal
Sync range that is beyond the capabilities of your monitor. If in doubt,
Choose a conservative setting.

    hsync in KHz; monitor type with characteristic modes
 1  31.5; Standard VGA, 640*480 @ 60Hz
 2  31.5 -  35.1; Super VGA, 800*600 @ 56Hz
 3  31.5, 35.5; 8514 Compatible, 1024*768 @ 87 Hz interlaced (no 800*600)
 4  31.5, 35.15, 35.5; Super VGA, 1024*768 @ 87 Hz interlaced, 800*600 @ 56 Hz
 5  31.5 - 37.9; Extended Super VGA, 800*600 @ 60 Hz, 640*480 @ 72 Hz
 6  31.5 - 48.5; Non-Interlaced SVGA, 1024*768 @ 60 Hz, 800*600 @ 72 Hz
 7  31.5 - 57.0; High Frequency SVGA, 1024*768 @ 70 Hz
 8  31.5 - 64.3; Monitor that can do 1280*1024 @ 60 Hz
 9  31.5 - 79.0; Monitor that can do 1280*1024 @ 74 Hz
 10 Enter your own horizontal sync range

Enter your choice (1-11):
```

그림 7-15. 수평동기범위차림표

자기의 수평동기범위에 맞는것을 찾을수 없어 수동으로 현시장치범위를 지정하는 경우 《Enter Your Own Horizontal Sync Range》를 선택해야 한다.이 정보를 호출하지 못하는 경우 보수적인 값을 선택할수 있다. 15in 다중동기현시장치에 대해서는 Super VGA 나 Extended Super VGA 를 선택해야 한다. 14in 현시장치에 대해서는 Standard VGA 를 선택하는것이 대체로 안전하다.

다음은 현시장치에 대한 수직동기범위를 설정한다. xf86config 는 많지 않은 항목을 주지만 《Enter Your Own Vertical Sync Range》의 선택을 리용할수 있다. 현시장치범위를 모르는 경우 가장 낮고 가장 작은 범위 즉 일반적으로 XFree86 의 대부분 판본에서는 50~70 으로 선택하는것이 가장 안전하다.

현시장치에 대한 자료를 일단 설정하면 xf86config 는 현시장치를 위한 자유로운 형식을 정의하겠는가 문의한다(실례로 My Linux Monitor). 이 정의는 XFree86 의 구성이나 성능에는 실제적으로 영향을 미치지 않지만 앞으로의 리용을 고려하여 XFree86 구성파일에 현시장치의 정의를 식별하는 간단한 방법을 준다.

일단 모든 현시장치정보가 입구되면 비데오기판이나 소편을 설정할수 있다. 첫번째 화면은 널리 알려 진 기판과 소편의 자료기지를 보겠는가고 문의한다. 이것을 선택하면 그림 7-16 에서와 같이 널리 알려 진 기판들의 차림표가 펼쳐 진다.

```
0 2 the Max MAXColor S3 Trio64V+               S3 Trio64V+
1 2-the-Max MAXColor 6000                      ET6000
2 3Dlabs Oxygen GMX                            PERMEDIA 2
3 3Dvision-i740 AGP                            Intel 740
4 3Dlabs Permedia2 (generic)                   PERMEDIA 2
5 928Movie                                     S3 928
6 ABIT G740 8MB SDRAM                  Intel 740
7 AGP 2D/3D V+ 1N, AGP-740D                    Intel 740
8 AGX (generic)                                AGX-014/15/16
9 ALG-5434(E)                                  CL-GD5434
10 Aopen AGP 2X 3D Navigator PA740             Intel 740
11 Aopen PA2010                                Voodoo Banshee
12 Aopen PA45                          SiS6326
13 Aopen PA50D                                 SiS6326
14 Aopen PA50E                                 SiS6326
15 Aopen PA50V                                 SiS6326
16 Aopen PA80/DVD                              SiS6326
17 Aopen PG128                                 S3 Trio3D

Enter a number to choose the corresponding card definition
Press enter for the next page, q to continue configuration.
```

그림 7-16. 비데오기판차림표

이 차림표에서 다음페지는 Enter 를 눌러서 넘어 간다. 기판을 선택하기 위해서 그 번호를 주고 Enter 를 누른다. 자기의 기판에 맞는것을 찾지 못하면 q 건을 눌러서 다음화면으로 넘어 간다.

비데오기판을 선택하면 차림표화면에는 그림 7-17 과 같은것이 현시된다. 여기에서 비데오기판의 비데오기억기를 선택해야 한다.

비데오기억기선택이 끝나면 비데오기판을 자유로운 형으로 지정하겠는가 문의하는 화면이 나타난다. 현시장치에서와 같이 이 항목들은 X Windows 체계의 고유한 기능들과는 무관계하며 자기의 XFree86 구성파일을 수동으로 편집할수 있다.

```
Now you must give information about your video card. This will be used for
the “Device” section of your video card in XF86Config.

You must indicate how much video memory you have. It is probably a good
idea to use the same approximate amount as that detected by the server you
intend to use. If you encounter problems that are due to the used server
not supporting the amount memory you have (e.g. ATI Mach64 is limited to
1024K with the SVGA server), specify the maximum amount supported by the
server.

How much video memory do you have on your video card:
 1  256K
 2  512K
 3  1024K
 4  2048K
 5  4096K
 6  Other

Enter your choice:
```

그림 7-17. 비데오기억기차림표

그림 7-18 에서 보여 준 다음차림표를 리용하면 매 화소당 색번호를 나타내는 색계조를 선택한다. 색계조비트가 커질수록 색이 고와진다. 실천적으로 리용할수 있는 색비트수는 도형처리기판의 기억기와 실행하고 있는 해상도에 따른다. 실례로 512KB 용량의 비데오기억기를 가진 도형처리기판에서는 24bit 색을 지원할수 없다.

```
Please specify which color depth you want to use by default:

 1    1 bit (monochrome)
 2    4 bits (16 colors)
 3    8 bits (256 colors)
 4    16 bits (65536 colors)
 5    24 bits (15 million colors)

Enter a number to choose the default depth.
```

그림 7-18. 색깊이차림표

마지막으로 XF86Config 파일을 어디에 작성하겠는가 하는 질문이 나타난다. 먼저 표준장소인 /etc/X11/XF86Config 에 파일을 작성하겠는가 묻는다. 이것은 낡은 파일에 덧쓰기되는것을 금지시키기 위해서이다. 이미 X Windows 를 리용하고 있었고 그것을 약간 다른 설정들로 재구성하려면 새것으로 바꾸기전에 xf86config 를 리용하여 자기의 낡은 구성파일을 복사하여 보관할수도 있다.

그 질문에 **n** 으로 대답하면 xf86config 는 다음절에서 보게 되는것처럼 파일을 쓸수 있는 다른 장소가 제공된다.

먼저 《표준》위치(/usr/X11R6/lib/X11/XF86Config)에 파일을 쓰겠는가고 묻는다. 파일에 대한 장소를 지정하려면 **n** 으로 대답한다.

다음 현행파일을 현행등록부에 쓰겠는가고 묻는다. 현재 등록부가 론리적인 장소(뿌리사용자의 홈등록부나 /tmp 와 같은)라면 **y** 로 대답할수 있다. **n** 으로 대답하는 경우 새로운 구성을 보관할수 있는 완전한 경로와 파일이름을 주라는 재촉문이 나타난다.

이 구성을 리용하기 위하여(그것이 만족하다고 생각된다면) XFree86 구성파일의 정확한 장소에 보관할 파일을 복사한다. XFree86 구성파일은 /etc/XF86Config 가 아니면

/etc/X11/XF86Config 이다. Red Hat 7.1 에서 그 파일은 /etc/X11/XF86Config 이다.

다른 방법은 xf86config 를 실행하기전에 여벌복사장소에 자기의 현행구성파일을 복사하는것이다. 즉 현행구성파일에 덧쓰기하겠는가고 물으면 간단히 **y** 로 대답한다. 제 14 장에서 파일을 어떻게 복사하는가 하는것을 배우게 되겠지만 다음의 지령을 리용하여 Red Hat 7.1 XFree86 구성파일의 여벌복사를 간단히 진행할수 있다.

```
$ cp /etc/X11/XF86Config /etc/X11/XF86Config.keep
```

이 지령이 실행되면 /etc/X11/XF86Config.keep 에 구성파일을 복사할수 있다.

구성검사

X Windows 의 구성이 끝나면 자기가 만든 구성이 동작하는가를 확인한다. 이를 위하여 X Windows 를 실행시킨다. Red Hat Linux 7.1 은 기정의 창문관리기와 일부 기정의 초기응용프로그람을 실행시키는 X Windows 와 함께 동작하므로 이제는 XFree86 을 구성할 준비가 되였다고 볼수 있다.

X Windows 를 기동하기 위하여 다음의 지령을 준다.

```
$ startx
```

이 지령에 의해서 X Windows 가 실행되기 시작한다. 순조롭게 기동되면 화면에 많은 정보행이 흐르며 계속하여 화면이 도형처리방식으로 절환되고 표준 X Windows 환경이 나타난다. 이것을 그림 7-19 에 보여 준다.

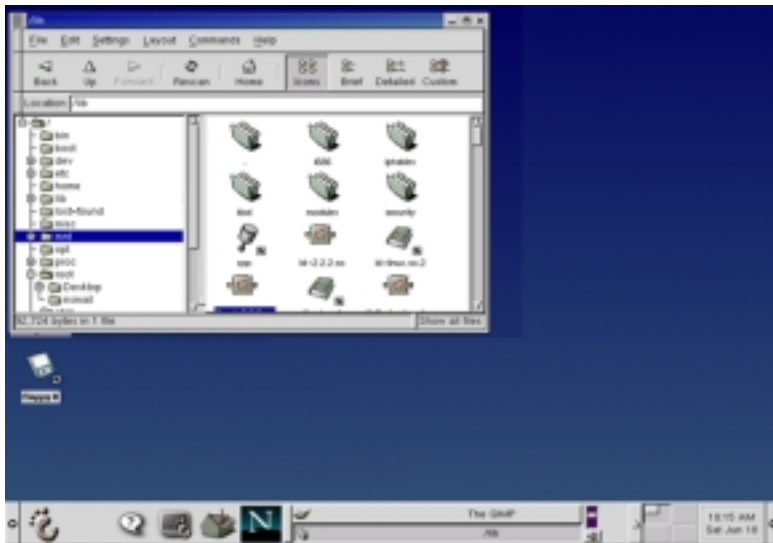



그림 7-19. Red Hat Linux 7.1 의 전형적인 탁상

X Windows 가 정확히 작업하고 있다는것을 확인하는것을 목적하는 정규적인 사용자나 뿌리사용자로서 그것을 기동할수 있다. 그러나 오직 뿌리사용자만이 수정할수 있는 중요한 체계파일을 우연히 파괴하고 삭제하고 변경시키는 X Windows 응용프로그람을 리용하는 기회를 될수록 없애기 위하여 비뿌리사용자로 X Windows 를 기동시킬것을 요구한다.

동작하지 않는 경우에는 문제가 있다. 실패는 보통 다음의 세가지 형태로 나타난다.

- 오유통보문이 현시되고 도형처리방식으로 절환하지 못하며 지령재촉문으로 돌아온다.
- 도형처리방식으로 절환되지만 탁상에 회색공간이 남아 있고 창문관리기가 없어지고 창문이 동작하지 못한다.
- 도형처리방식으로 절환되지만 잠간 생기고 지령재촉문으로 돌아 간다.

처음의 경우에는 XFree86 이 정확하게 구성되지 못했다고 보는것이 적합하다. 공교롭게도 가능한 구성실패원인의 수는 존재하는 하드웨어의 결합수만큼 많을것이다.

Xconfigurator 나 xf86config 를 다시 실행하기전에 작업하는 도구를 가지도록 하는 몇가지 조언을 참고해야 한다.

- 자기의 비데오기판과 현시장치에 대한 자료를 재확인해야 한다.
- 리용할수 있는 더 작은 수평 및 수직동기범위를 시험해 보아야 한다.
- SVGA 봉사기대신에 VGA16, 전용화된 가속기판봉사기대신에 SVGA 봉사기와 같은 더 간단한 봉사기의 리용을 시험해 보아야 한다.

이 시도들이 모두 실패하는 경우 제기되는 문제점을 조사해 보아야 한다. 가장 좋은 방법은 XFree86 HOWTO 파일을 읽어 보는것이다. 비교적 완전한 배포물이 있다면 이미 체계에 이것이 설치되여 있을수도 있다. 그렇치 않으면 http://www.linuxdoc.org/HOWTO/XFree86-HOWTO/index.html 에 있는 Web 의 HOWTO 를 조사할수 있다. 자기의 XFree86 판본에 맞는 문서를 선택하여 조사할수 있다. 4.0.3 판본에 대해서는 http://www.xfree86.org/4.0.3/index.html 에서 참고하면 된다.

두번째와 세번째의 경우에는(도형처리방식으로 절환되지만 동작하지 않는다.) 우선 X Windows 기동구성과 창문관리기구성을 검색한다. 제 12 장에서는 이러한 구성을 상세하게 고찰한다.

▶미리 알아두기

지금까지 X Windows 를 기동시키기 위한 작업을 진행하였다. 이제부터는 실천적인 과제들에 대한 X Windows 환경을 구성해야 한다. 이를 위하여 GNOME 즉 GNU Object Model Environment 를 보자.

GNOME 은 앞으로 출현하게 될 Linux 도형처리탁상을 위한 경쟁적인 두 표준들중의 하나이다. GNOME 은 응용프로그람을 개발하고 실행시키기 위해서 완전히 통합된 탁상환경이다. GNOME 은 Red Hat Linux 7.1 의 표준탁상이다.

제 8 장 《GNOME 과 X Windows 의 리용》에서는 GNOME 탁상에서 X Windows 를 리용하는 기초를 고찰한다. 제 9 장 《GNOME 과 X Windows 에서 응용프로그람의 리용》에서는 말단모방기, 파일관리기, 도형응용프로그람 등을 비롯하여 GNOME 과 X Windows 응용프로그람의 실행을 고찰한다. 이 두개 장을 거치면 다른 리용가능한 X Windows 응용프로그람을 실험해 볼수 있는 충분한 경험과 확신성을 가지게 된다.

제 8 장. GNOME 과 X Windows 의 리용

지금까지 체계에 X Windows 를 설치하고 기동시켰다. X Windows 가 Linux 를 위해 설계된것만큼 X Windows 환경을 리용하는 방법을 배워야 한다.

이 장에서는 GNOME 탁상환경의 리용방법을 고찰한다. GNOME 탁상은 매우 복잡하므로 그자체가 자기자신에 대한 설명서로도 된다. 이 장의 목적은 GNOME 리용에 대한 완전한 설명서를 주는것이 아니라 GNOME 을 원만히 쓸수 있게 하는데 있다. Windows 9x, Me, NT 나 2000 을 써보았으면 GNOME 의 기초적인 내용을 쉽게 리해할수 있다.

X Windows 와 GNOME 의 기동

Red Hat 는 GNOME 으로 구성되여 있기때문에 탁상관리기를 기동하는것은 X Windows 환경을 기동하는것만큼 간단하다. 제 7 장 《X Windows 의 설치와 구성》에서 고찰한바와 같이 사용자로 가입하고 다음의 지령을 준다.

```
$ startx
```

이 지령을 주면 먼저 X Windows 가 기동하고 다음에 GNOME 탁상관리기가 기동한다. 초기화면은 그림 8-1 과 같다. 여기에는 두가지 형태의 아이콘 즉 응용프로그람을 실행시키는 기동프로그람과 GNU Midnight Commander 로 알려 진 폴더가 있는데 이 폴더는 GNOME 파일관리기로 연다.

주의	Red Hat 는 체계를 기동할 때 Red Hat 를 어떻게 설치하는가에 따라 도형방식가입재촉문이 나오면 X Windows 를 기동시킬수 있다. 이 경우에는 가입하자마자 GNOME 이 즉시에 자동적으로 실행되므로 **startx** 지령을 리용할 필요가 없다.

수동으로 GNOME 을 설치

Red Hat Linux 7.1 은 기정으로 GNOME 탁상환경을 설치한다. 만일 어떤 리유로 GNOME 설치를 선택하지 않았거나 GNOME 이 자동적으로 설치되지 않으면 GNOME 을 수동으로 설치하여야 한다.

이를 위해서 Red Hat 설치 CD-ROM 의 RedHat/RPMS 등록부로부터 다음과 같은 프로그람묶음을 설치해야 한다.

- gnome-applets-1.2.4-3.i386.rpm
- gnome-audio-1.0.0-12.noarch.rpm
- gnome-audio-extra-1.0.0-12.noarch.rpm
- gnome-core-1.2.4-16.i386.rpm
- gnome-core-devel-1.2.4-16.i386.rpm
- gnome-games-1.2.0-10. i386.rpm
- gnome-games-devel-1.2.0-10.i386.rpm
- gnome-kerberos-0.2.2-2.i386.rpm

- gnome-libs-1.2.8-11.i386.rpm
- gnome-libs-devel-1.2.8-11.i386.rpm
- gnome-linuxconf-0.64-1.i386.rpm
- gnome-lokkit-0.43-6.i386.rpm
- gnome-media-1.2.0-12.i386.rpm
- gnome-objc-1.0.2-11.i386.rpm
- gnome-objc-devel-1.0.2-11.i386.rpm
- gnome-pim-1.2.0-9.i386.rpm
- gnome-pim-devel-1.2.0-9.i386.rpm
- gnome-print-0.25-9.i386.rpm
- gnome-users-guide-1.2-3.noarch.rpm
- gnome-utils-1.2.1-5.i386.rpm

이 묶음의 일부는 두번째 Red Hat 설치 CD-ROM 에 있다. 이 책과 함께 제공된 Red Hat 7.1 판본만이 있다면 불필요한 다른 프로그람묶음까지 설치할 필요는 없다. 그 프로그람묶음은 GNOME 의 기본조작에서 본질적인것이 못되므로 걱정하지 않아도 된다. 이 프로그람묶음을 설치하자면 rpm 지령을 리용해야 한다. 즉

 # rpm –i 파일이름

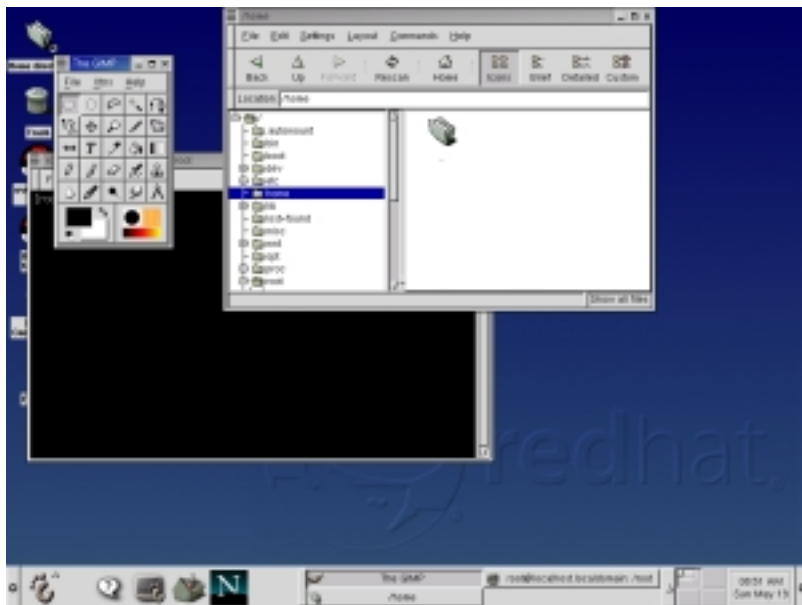



그림 8-1. Red Hat Linux 7.1 의 전형적인 GNOME 탁상

GNOME 조작판

그림 8-2 에 보여 준 GNOME 조종판은 화면아래에 가로방향으로 현시된다. GNOME 조종판에는 체계차림표단추(제일 왼쪽에 발바닥모양으로 그려 진 단추), 제일

오른쪽에 시간과 날자를 표시하는 작은 조종판과 표준동작을 나타내는 종이와 기타 단추들이 포함된다.

그림 8-2. GNOME 조종판

이 조종판에는 특수한 프로그람을 실행하는 기동도구로 봉사하는 5 개의 단추가 있다. 즉

- 열쇠단추는 통과암호로 보호된 화면보호기를 기동시킨다.
- 물음표식단추는 GNOME 도움말체계를 기동시킨다.
- 현시장치단추는 GNOME 지령행말단을 기동시킨다.
- 도구함단추는 GNOME 조종쎈터를 기동시킨다.
- N 표식단추는 현재 Netscape 로 널리 알려 진 기호이다.

체계차림표의 리용

체계차림표는 Windows 98 과 Windows 2000 의 start 차림표와 같이 동작한다. 체계차림표단추를 누르면 그림 8-3 과 같은 차림표가 현시된다.

주의 이 장에서 보여 준 그림들은 Red Hat 7.1 의 표준 GNOME 설치에서와 같은 시감각을 가지지는 않지만 내용은 같다.

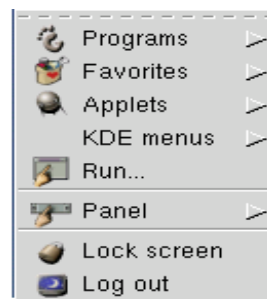



그림 8-3. GNOME 체계차림표

체계차림표는 여러개의 항목을 포괄하고 있다.

Programs 이 보조차림표를 리용하면 GNOME 에서 쓸수 있는 대부분의 프로그람과 도구를 호출할수 있다.

Favorites 이 선택항목을 리용하여 Web 싸이트와 같은 자기가 작성한 즐겨 찾는 항목을 호출할수 있다.

Applets 이 보조차림표를 리용하여 현시장치, 시계, CD 재생기들을 비롯하여 표준 GNOME 애플레트를 호출할수 있다.

KDE Menus 자기의 체계에 대리탁상관리기로 KDE 를 설치한 경우 이 보조차림표를 리용하여 표준 KDE 차림표를 호출할수 있다.

Run 이 선택항목을 리용하여 Linux 체계에 설치된 임의의 프로그람과 응용프로그람을 실행시킬수 있다. 이 항목을 선

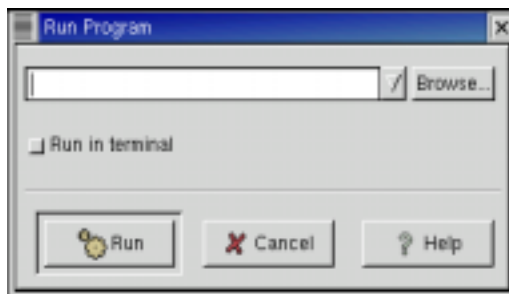



그림 8-4. 프로그람의 실행

택하면 그림 8-4 와 같은 대화칸이 펼쳐 진다. 프로그람의 경로와 이름을 입력하고 Run 단추를 찰칵하면 응용프로그람이 실행된다. 또는 그림 8-5 와 같은 파일선택창문으로부터 프로그람을 선택하기 위하여 Browse 단추를 찰칵할수 있다.

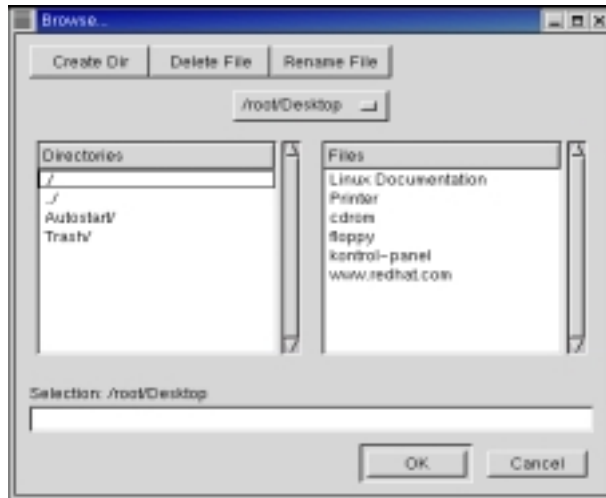



그림 8-5. 실행하려는 프로그람의 선택

Panel 이 보조차림표는 GNOME 조종판을 구성하는 도구를 제공한다. 제 10 장 《고급한 GNOME 구성》에서 더 상세하게 고찰한다.

Lock Screen 이 선택항목을 리용하여 통과암호를 가진 화면보호기를 기동시킬수 있다.

Log Out 이 선택항목을 리용하여 GNOME 의 상태를 기록하고 끝낼수 있다. 만일 도형방식가입화면에 가입하여 GNOME 를 기동했으면 이 선택항목은 GNOME 에서 탈퇴하여 가입화면으로 돌아 간다. 지령행조종탁에서 GNOME 을 기동했으면 지령행상태로 돌아 간다.

Program Submenu Program 보조차림표에서 호출할수 있는 여러개의 추가적인 보조차림표가 있는데 매 보조차림표이름의 오른쪽에는 오른쪽방향화살표식이 있다. 이 매개의 보조차림표들은 실행할수 있는 프로그람이 들어 있는 보조차림표를 연다. 표준보조차림표들은 다음과 같다.

Applications 이 보조차림표는 GNOME 달력도구, GNOME 표처리프로그람, Gnumeric 와 같은 표준 GNOME 응용프로그람을 제공한다.

Utilities 이 보조차림표를 리용하여 말단, 전자수판, 파일탐색도구를 비롯하여 GNOME 과 X Windows 에서 널리 쓰이는 편의프로그람을 호출한다.

Development 이 보조차림표에는 LADE, GTK+ 사용자대면부작성도구로 불리우는 개발편의프로그람이 들어 있다.

Games X Windows 와 GNOME 유희프로그람을 제공한다.

Graphics 이 보조차림표는 호평 받는 영상편집프로그람 The GIMP 를 비롯하여 GNOME 의 도형편집도구를 제공한다.

Internet 이 보조차림표를 리용하면 Netscape, 새로운 Mozilla 열람기, GNOME FTP 도구인 gFTP 를 비롯하여 표준적인 인터네트도구들을 빨리 호출할수 있다.

Multimedia 이 보조차림표를 리용하면 CD 재생기와 음량조절기를 비롯하여 다

매체도구들을 선택할수 있다.

Settings 이 보조차림표는 GNOME 구성을 위한 도구묶음을 제공한다.

System 이 보조차림표는 Red Hat rpm 프로그람제품인 GnoRPM 을 위한 GNOME 방식프로그람제품관리체계를 비롯하여 몇가지 체계도구에 대한 호출을 제공한다.

File Manager 이 차림표항목을 선택하여 그림 8-6 에 보여 준것과 같은 GNOME 파일관리기를 기동한다. GNOME 파일관리기에 대해서는 후에 상세하게 고찰한다.

Help System 이 선택항목은 GNOME 도움말체계를 기동하는데 리용된다. 도움말체계는 그림 8-7 에 보여 준바와 같이 HTML 열람기를 리용하고 있다. GNOME 도움말체계는 GNOME 의 리용에 대한 포괄적인 문서들을 제공하므로 GNOME 리용의 보다 상세한 내용들에 대해서는 그것을 참고할수 있다.

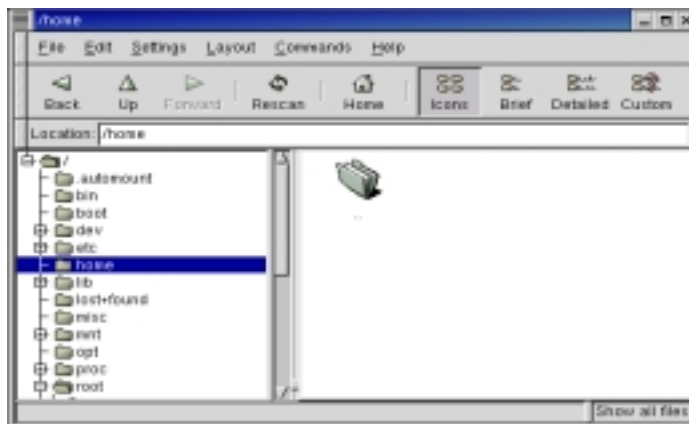



그림 8-6. GNOME 파일관리기

그림 8-7. GNOME 도움말체계

GNOME 조종판의 조종

GNOME 조종판의 한가지 결함은 휴대형콤퓨터나 보다 작은 14 인치 혹은 15 인치현

시장치를 가진 콤퓨터들에서처럼 작은 화면에서 작업하는 사용자들에게는 상당한 정도의 화면공간을 차지한다는것이다. 다행히 이 문제를 해결할수 있는 몇가지 방법이 있다.

하나는 조종판을 작은 4 각형으로 만드는것이다. 표준조종판은 화면아래에 가로 놓여 있다. 조종판의 왼쪽 끝에 있는 작은 왼쪽방향화살단추를 찰칵하여 화면의 아래 왼쪽구석에 있는 작은 4 각형단추에 조종판을 함축시킬수 있다. 그러면 조종판은 보이지 않고 오른쪽화살표식이 있는 작은 4 각형단추만 남게 된다. 오른쪽화살단추를 찰칵하면 조종판이 다시 펼쳐 져 화면아래부분이 차게 된다.

같은 방법으로 조종판의 오른쪽 구석에 있는 오른방향화살단추를 찰칵하면 왼쪽방향화살단추만 남고 조종판은 화면의 아래오른쪽구석에 함축된다. 왼쪽방향화살단추를 찰칵하면 조종판이 다시 펼쳐 진다.

이것도 불편하고 화살단추가 아직도 많은 공간을 차지한다고 생각되면 단추에서 화살표를 소거하여 함축된 조종판이 완전히 자리를 차지하지 않도록 할수 있다. 이를 위해 조종판의 배경을 오른쪽찰칵한다. 나타나는 차림표에서 Panel → Properties → All Properties 를 선택한다. 그러면 그림 8-8 에 보여 준 조종판에 대한 Panel Properties 대화칸이 펼쳐 진다.

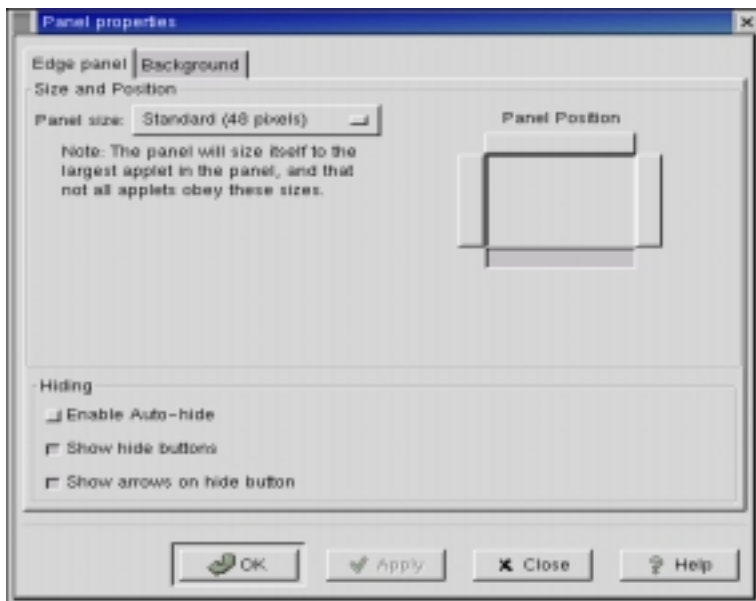



그림 8-8. GNOME 의 Panel Properties 대화칸

여기서 Show Arrows On Hide Button 선택항목을 해제하고 Apply 단추를 찰칵하면 조종판의 끝에 있는 단추에서 왼쪽 혹은 오른쪽화살단추를 삭제할수 있으며 조종판을 될수록 좁게 만들수 있다.

탁상공간을 효과적으로 리용하기 위한 또 다른 선택항목은 Panel Properties 대화칸에서 Enable Auto-hide 선택항목을 선택하고 Apply 단추를 찰칵하는것이다. 이렇게 하면 조종판이 숨겨 지고 화면아래에 가로 놓인 좁은 띠만이 보인다. 마우스가 이 좁은 띠우에서 움직일 때만 전체 조종판을 볼수 있다. 마우스지시자가 조종판을 벗어 나는 순간 조종판은 다시 자동적으로 숨겨 지고 작은 띠만이 보인다.

Panel Properties 대화칸에서 다른 한가지 좋은 점은 늘 조종판을 아래에만 배치하도록 하지 않고 화면의 임의의 구석에 배치할수 있게 한것이다. 례를 들어 어떤 사용자들은 조종판을 우에, 왼쪽에 혹은 오른쪽에 놓는것을 더 좋아한다. Panel Position 항목의 대화칸에서 적당한 방향을 선택하고 Apply 단추를 찰칵하면 된다.

조종판애플레트

GNOME 조종판은 아주 유연한 도구이다. 그것의 제일 큰 우점은 조종판의 작은 애플레트를 조종하는것이다.

GNOME 에는 많은 애플레트가 있으며 프로그람작성자들이 자기식의 애플레트를 만들수 있는 대면부를 제공한다. Red Hat Linux 7.1 의 GNOME 에 있는 애플레트는 다음과 같다.

- 작은벽돌쌓기유희도구
- 축전지리용, CPU 적재, 디스크공간을 추적하기 위한 현시장치들
- CD 재생애플레트
- 새 통보문들에 대한 수신함을 검사하는 애플레트
- 시계애플레트
- 인쇄기조종기
- 가상탁상을 제공하기 위한 탁상안내애플레트

기정으로 시계와 탁상안내애플레트들은 GNOME 조종판에 현시된다.
조종판애플레트로 작업할 때 여러가지 동작을 하게 된다.

- 조종판에 애플레트의 추가
- 조종판에서 애플레트의 옮기기
- 애플레트속성의 변경
- 조종판에서 애플레트의 삭제

조종판에 애플레트의 추가

조종판에 애플레트를 추가하는것은 아주 쉽다. Desk Guide 애플레트를 추가하려면 조종판의 배경단추를 찰칵해야 한다. 그러면 차림표가 나타난다. 이 차림표에서 Panel → Add To Panel → Applet → Utility → Desk Guide 를 선택한다. 결과 조종판에 4 개의 블로크묶음이 나타난다.

기구를 추가하는 다른 한가지 방법은 기본 System 차림표로부터 하는 방법이다. Tasklist 기구를 추가하기 위하여 기본 System 차림표를 찰칵하고 Applets → Utility → Tasklist 를 선택한다. 그러면 조종판에 현재 실행되고 있는 응용프로그람들을 목록화한 Tasklist 가 나타난다.

조종판에서 애플레트의 옮기기

조종판에 있는 어떤 애플레트를 그 조종판의 다른 위치로 옮기려고 할수 있다. 애플레트를 옮기려면 그 애플레트를 찰칵하고 현시된 차림표에서 Move 를 선택해야 한다. 그다음 조종판의 새 위치에로 마우스로 애플레트를 끌기한 다음 찰칵하면 결국 그 위치로 옮겨 진다.

애플레트속성의 변경

대부분의 애플레트들은 자기의 동작을 선택할수 있는 속성들을 가지고 있다. 이 속성들은 애플레트를 찰칵하여 현시되는 차림표에서 Properties 를 선택하여 펼쳐 지는 대화칸에서 조종한다.

조종판에서 애플레트의 삭제

일단 애플레트를 조종판에 배치했으면 그것을 삭제할줄도 알아야 한다. 이것으로 해서 GNOME 에 있는 많은 애플레트를 쉽게 리용하고 조종판이 너무 란잡해 지면 조종판을 깨끗하게 할수도 있다. 애플레트를 삭제하려면 그것을 오른쪽 마우스단추로 찰칵할때 현시되는 차림표에서 Remove From Panel 을 선택해야 한다.

Desk Guide 및 Tasklist 애플레트의 리용

이 절에서는 이미 추가한 Desk Guide 및 Tasklist 애플레트를 설명하는것으로 애플레트를 리용하는 방법을 본다. 그림 8-9 에 보여 준 Desk Guide 애플레트는 가상탁상을 제공한다.

주의	Desk Guide 및 Tasklist 애플레트는 GNOME 의 이전판본들에서 Pager 로 알려 져 있다.

그림 8-9. Desk Guide 및 Tasklist 애플레트

가상탁상의 개념은 간단하다. 간단한 실례를 들어 보자. 만일 두개의 가상탁상이 있다면 이것은 현시장치가 둘이지만 그 현시장치들에 대한 전원케블은 하나인것처럼 생각하면 된다. 현시장치에서 열리고 실행되며 현시되는 서로 다른 두조의 응용프로그람을 가질수 있으며 현시장치사이 전원케블을 절환하면 매 환경에로 절환할수 있다.

가상탁상은 류사한 기능을 제공하는데 두 현시장치사이의 전원케블을 절환하기가 힘들거나 실패하거나 지연되는 현상은 없다. 가상탁상체계에서 매 탁상은 창문을 열고 응용프로그람을 실행할수 있는 새로운 작업환경이다. 동시에 어느 탁상에 나타나든지 관계없이 이 응용프로그람들은 같은 X Windows 대화시간에 실행되는데 이것은 응용프로그람들이 실제로 같은 탁상에 현시된것과 같이 그것을 가르거나 붙이기할수 있다는것을 의미한다.

Desk Guide 의 4 개 4 각형단추가운데서 임의의 단추에 마우스를 대고 찰칵하여 탁상을 절환할수 있다. 그러면 즉시 선택된 탁상에로 절환되며 그 탁상이 펼쳐 진다.

한편 Tasklist 애플레트는 현재 탁상에 펼쳐 진 모든 창문들에 대한 단추를 현시하는 구역을 차지한다. 이 단추들을 찰칵하여 임의의 창문을 전경에 전개할수 있다. 만일 창문이 이미 전경에 있으면 Tasklist 단추를 다시 찰칵하여 최소화할수 있다.

그림 8-9 를 보면 응용프로그람단추들과 가상탁상사이에는 웃방향건이 있는 작은 단추가 있다. 이 단추를 찰칵하면 모든 가상탁상들에 펼쳐 진 전체 창문들에 대한 차림표가 현시된다. 응용프로그람을 찰칵하여 창문이 있는 탁상에로 절환하고 전경에로 창문을

가져 올수 있다.

GNOME 대면부

GNOME 에는 많은 중요한 대면부요소가 있다. 이 절에서는 창문을 관리하고 탁상에서 작업할수 있게 하는 GNOME 의 특성들에 대하여 고찰한다.

창문조종

GNOME 의 창문은 Windows 98 과 대체로 비슷하며 거의 같은 방법으로 동작한다. 이것은 Windows 사용자들에게 매우 편리하다. 왜냐하면 그것이 Microsoft Windows 조작체계로부터 X Windows 에로의 이동을 창문과의 호상작용에 관한 간단한 과정으로 만들기때문이다.

기본창문구성요소들 즉 제목띠, 창문조작아이콘들, Window Operations 차림표, 창문경계를 보자.

제목띠(title bar)

제목띠는 창문의 맨 우에 가로 놓인다. 제목띠에는 창문이름이 표시되는데 대체로 창문을 현시하는 응용프로그람의 이름이 표시된다. 가령 xterm 창문인 경우 제목은 기정으로 xterm 이다(이것은 변경시킬수 있다.이에 대해서는 후에 보기로 한다.).

GNOME 에서 제목띠구역은 몇가지 기능을 제공한다. 제목띠를 마우스로 찰칵하면 열려 져 있는 다른 창문들의 앞으로 필요한 창문을 내올수도 있다. 마우스의 왼쪽단추를 누르고 끌면서 창문을 마음대로 움직일수 있다. 마우스의 왼쪽단추를 놓으면 창문은 새 위치에 놓인다.

제목띠를 두번찰칵하면 창문이 축소되여 제목띠만 보이게 된다. 제목띠를 다시 두번 찰칵하면 창문내용이 다시 펼쳐 진다. GNOME 전문용어로 이것을 창문의 은폐 및 비은폐(shading and unshading)라고 한다.

창문작용아이콘

대부분의 창문에는 오른쪽 웃구석에 세개가 하나의 그룹인 창문작용아이콘이 현시된다. 왼쪽으로부터 오른쪽으로 가면서 이 아이콘의 기능을 고찰한다.

창문의 최소화 이 아이콘은 창문을 GNOME 조종판의 Tasklist 애플레트단추로만 남기고 숨긴다.

창문의 최대화/재설정 이 아이콘은 창문을 최대화하거나 그전의 본래크기로 절환한다.

창문의 닫기 이 아이콘은 창문을 닫는다. 여러개의 창문을 펼치고 작업하는 응용프로그람인 경우에는 필요한 창문만 닫고 다른 창문은 펼쳐 진 상태로 남겨 둔다.

Window Operation 차림표

제목의 왼쪽, 창문의 왼쪽웃구석에는 찰칵하면 Start 차림표로부터 Window Operation 차림표를 현시하는 아이콘이 있다. 이 차림표는 창문을 조작하는 몇가지 기능을 제공한다. 첫 세개의 기능은 앞에서 서술한 아이콘과 같은 최소화, 최대화, 닫기기

능이다. 기타 기능들은 다음과 같이 분류되여 보조차림표에서 찾을수 있다.

Toggle 이 항목에 속하는 보조차림표들은 탁상을 변경하였을 때 현재창문이 그대로 남아 있게 하던가 아니면 축소되게 하던가 또는 제목띠만 남아 있게 한다.

In Group 이 보조차림표는 현재의 창문을 다른 창문에 결합시킨다.

Send Window To 이 보조차림표를 리용하여 창문이동방향을 정해 줌으로써 현재의 창문을 다른 탁상에로 옮길수 있다(례를 들어 현재창문의 오른쪽창문).

Stacking 이 보조차림표는 현재창문이 다른 열린 창문들에 적절하게 《탄창화》된 장소를 결정한다.

Frame Type 이 보조차림표는 제목띠 혹은 창문테두리의 어디에 프레임을 배치하겠는가를 결정한다.

Frame Style Frame Style 차림표의 보조차림표들은 프레임의 시감각을 결정한다.

History 이 보조차림표들은 기본창문의 기정위치, 크기와 속성을 변경시킨다.

창문경계

대부분의 창문들은 5 화소폭의 경계를 가진다. 이 경계는 창문의 크기를 변경하기 쉽게 창문의 경계를 시각적으로 나타내는데 쓰인다.

창문의 크기를 수직, 수평으로 혹은 동시에 두방향으로 변경시킬수 있다. 제목띠와 직접 린접한 변두리를 제외한 경계의 아무 곳을 찰칵하고 끌기한다. 현재의 창문경계밖으로 끌기하면 창문은 마우스지시자의 위치에 따라 크기가 변한다. 왼쪽 혹은 오른쪽경계를 찰칵하면 수평으로 변경되며 아래경계를 찰칵하면 수직으로 변경된다. 그리고 아래구석을 찰칵하면 수평, 수직 두 방향으로 크기를 동시에 변경시킬수 있다.

GNOME 탁상

처음에 GNOME 을 기동하였을 때 GNOME 탁상에 한개의 폴더아이콘이 있는것을 보았을것이다. 이 아이콘은 Home Directory 라는 이름을 가진 서류철이다. 이 아이콘을 두번찰칵하면 GNOME 파일관리기가 열리고 그림 8-10 과 같이 파일관리기에서 현재 사용자의 홈등록부가 현시되는것을 보게 된다.

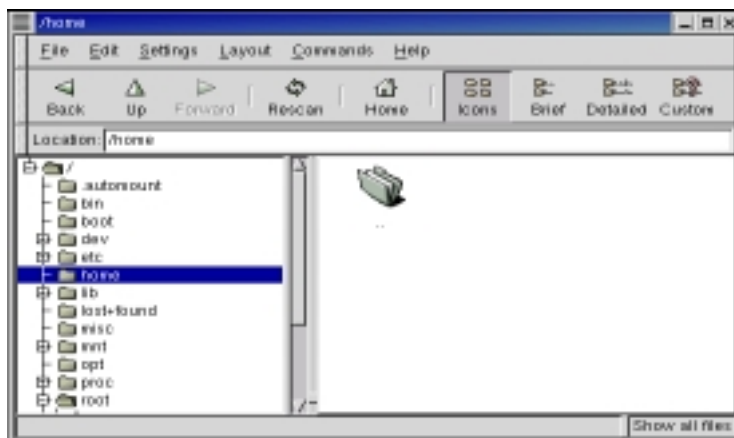



그림 8-10. GNOME 파일관리기

GNOME 탁상을 찰칵하면 여러가지 목적에 리용할수 있는 다음과 같은 차림표가 나타난다.

- 새 GNOME 파일관리기창문의 열기
- 탁상의 아이콘정돈
- 새 말단창문의 열기

례를 들어 탁상에서 Arrange Icons 를 선택하면 아이콘들이 살창모양으로 정확히 재배치된다. 이 기능은 탁상우에 많은 아이콘들이 란잡하게 불규칙적으로 배치되여 있을때 쓸모 있다. Create New Window 항목은 새 파일관리기창문을 연다(탁상에서 Home Directory 폴더를 두번찰칵하여 여는것과 같다.).

GNOME 파일관리

Windows 3.1 판본을 리용하는 사용자들은 File Manager 를, Windows 95/98 을 리용하는 사용자들은 Windows Explorer 를, Mac OS 를 리용하는 사용자들은 Finde 를 잘 알고 있다. 이 모든 편의프로그람들은 파일을 관리(복사, 옮기기, 이름바꾸기, 삭제 등)하고 응용프로그람을 실행시키는 특성들을 제공한다.

X Windows 의 세계에는 전통적으로 이 기능을 수행하는 표준도구가 없었다. 파일관리기 혹은 탁상관리기라고 하는 수많은 응용프로그람들은 이 기능을 제공하려고 한다. http://www.xnet.com/~blatura/linapp2.html#file에서 파일관리기의 목록을 찾을수 있다.

그러나 Linux 의 표준탁상환경으로서 GNOME 과 KDE 가 출현함으로써 그러한 현상은 없어 졌다. GNOME 과 KDE 는 둘 다 표준적인 파일관리기를 제공한다. GNU Midnight Commander 라고도 알려 진 GNOME 파일관리기는 유연하고 성능이 높은 파일관리도구와 탐색도구를 제공한다. 이 절에서는 GNOME 파일관리기에 대하여 고찰한다

GNOME 파일관리기의 기동

GNOME 파일관리기는 기정 GNOME Programs 보조차림표로부터 File Manager 를 선택하여 기동시킬수 있다. 다르게는 xterm 창문의 지령행에서 다음의 지령

```
$ gmc &
```

을 주거나 또는 완전한 경로를 지정하는 경우에는 Led Hat Linux 7.1 설치에서

```
$ /usr/bin/gmc &
```

를 정의하면 된다. 여기서 gmc 는 GNU Midnight Commander 를 의미한다.

파일관리기창문

그림 8-10 에서 이미 본것과 같이 파일관리기는 콤퓨터의 어느 한 등록부의 내용을 현시한다. 현시되는 등록부는 기정으로 프로그람을 실행하고 있는 사용자의 홈등록부이다. 그림에서 사용자 mj 가 파일관리기를 실행하였으므로 현시된 등록부는 사용자의 홈

등록부 즉 /home/mj이다.

창문은 세개의 기본부분으로 나누어 져 있다.

- 등록부나무(왼쪽)
- 현시부(오른쪽)
- 도구띠(우에 가로 놓인다.)

등록부나무

GNOME 파일관리기창문의 왼쪽에는 확장하거나 접을수 있는 Linux 체계의 등록부가 있다. 이 등록부나무는 Windows 95, 98, 2000 의 Windows Explorer 를 련상시킨다. 이 나무는 그 체계에서와 같은 방법으로 동작한다.

- 등록부이름옆의 오른쪽으로 향한 3 각형을 찰칵하면 부분나무가 확장되면서 그 등록부 다음준위의 보조등록부들이 현시된다.
- 등록부이름옆의 아래로 향한 3 각형을 찰칵하면 부분나무는 없어 지고 모든 자식등록부들이 감추어 진다.
- 등록부의 이름을 찰칵하면 파일관리기창문의 오른쪽에 등록부의 내용들이 현시된다.

주의 기정 Red Hat Linux 7.1 를 설치할 때 제공되는 판본을 비롯하여 일부 GNOME 판본에는 그림 8-10 에서 본 오른쪽 및 아래로 향한 3 각형의 자리에 +기호와 –기호가 놓여 있다.

현시부

GNOME 파일관리기의 오른쪽에는 현재 선택되여 있는 등록부나무의 내용을 현시하는 현시부가 있다. 이 현시부에서 표준현시는 매 파일을 완전한 크기를 가진 아이콘으로 보여 준다. 각이한 형태의 파일들과 등록부와 같은 기타 요소들을 표시하는 여러가지 형태의 아이콘들이 있다. 널리 알려 진 아이콘은 등록부, 일반문서, 프로그람묶음과 문서고 그리고 HTML 파일에 대한 아이콘이다.

주의 파일형태와 등록부구조에 대한 완전한 내용은 제 14 장 《파일다루기》에서 고찰한다. 이미 제 4 장 《Red Hat Linux 7.1 의 설치》에서 제공된 고찰로부터 Linux 등록부구조의 기초적인 개념은 가졌을것이다.

폴더아이콘은 등록부를 표시하며 Macintosh 와 Windows 체계사용자들에게 친숙해 진것이다.

종이쪼각아이콘은 text 파일이나 Microsoft Word 파일과 같은 일반문서들을 표시한다.

우에 떠 있는 HTML 단어를 가진 종이쪼각아이콘은 HTML 파일을 표시한다.

마지막으로 프레스아이콘은 Zip 파일과 같은 프로그람묶음과 문서고들을 표시한다.

도구띠

파일관리기창문의 우에 가로 놓인것은 9 개의 단추들이 있는 도구띠이다.

Back 이전에 현시되였던 등록부로 돌아 간다.

Up 현행등록부의 어미등록부로 이동한다.

Forward 기록렬의 다음등록부로 돌아 간다(앞으로 이동할수 있게 하자면 Back 단추를 리용하여 뒤로 이동해야 한다.)

Rescan 현재 현시된 등록부를 다시 조사하고 현시된 파일들을 디스크에 정확히 기록된 파일들과 동기시킨다. 현시장치는 다른 프로그람이 현재 현시된 등록부의 내용에 대한 바꾸기를 진행할 때 자동적으로 갱신되지 않는다. 그러므로 파일관리기창문에 이 변화를 반영하기 위하여 Rescan 을 리용해야 한다.

Home 사용자의 홈등록부로 옮겨 간다.

Icons 현재 현시방식을 큰 아이콘방식으로 절환한다(이것은 GNOME 파일관리기를 열었을 때의 표준방식이다.).

Brief 현재 현시방식을 작은 서류철아이콘과 그다음에 등록부이름을 쓰는 상세목록으로 절환한다(그림 8-11).

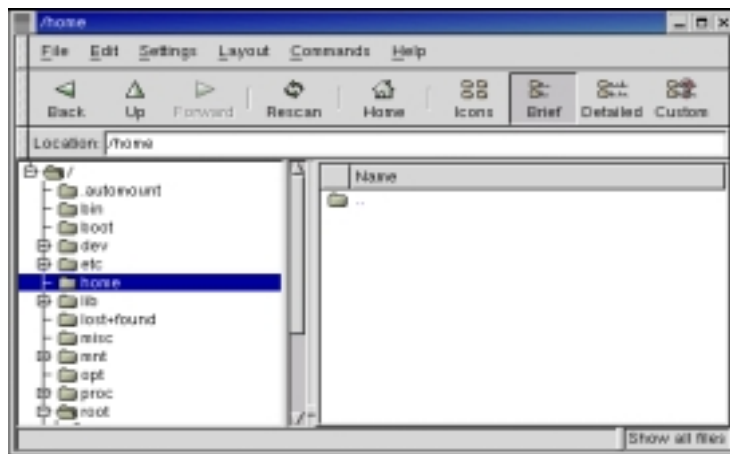



그림 8-11. Brief 방식으로 등록부현시

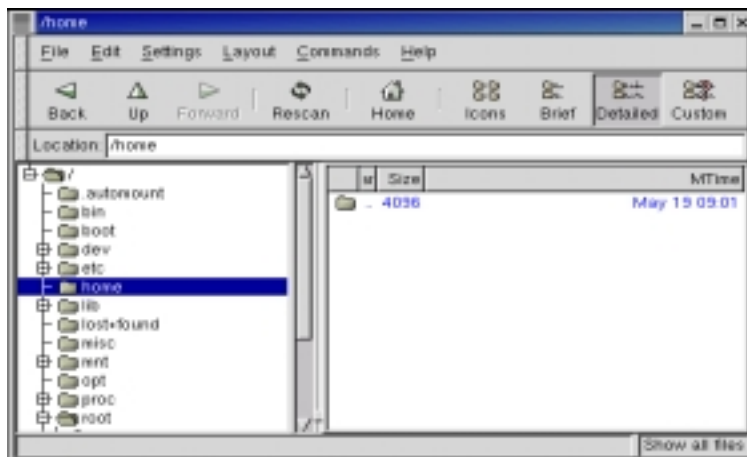



그림 8-12. 상세한 방식으로 등록부를 현시

Detailed 현재 현시방식을 이름, 크기, 파일 혹은 폴더의 변경시간이 있는 상세한 목록으로 바꾼다(그림 8-12 에 보여 준다.).

Custom 현재 현시방식을 사용자가 정의한 전용화된 통계정보를 현시하는 상세한 목록으로 절환한다(그림 8-13 에서 보여 준것과 같이).

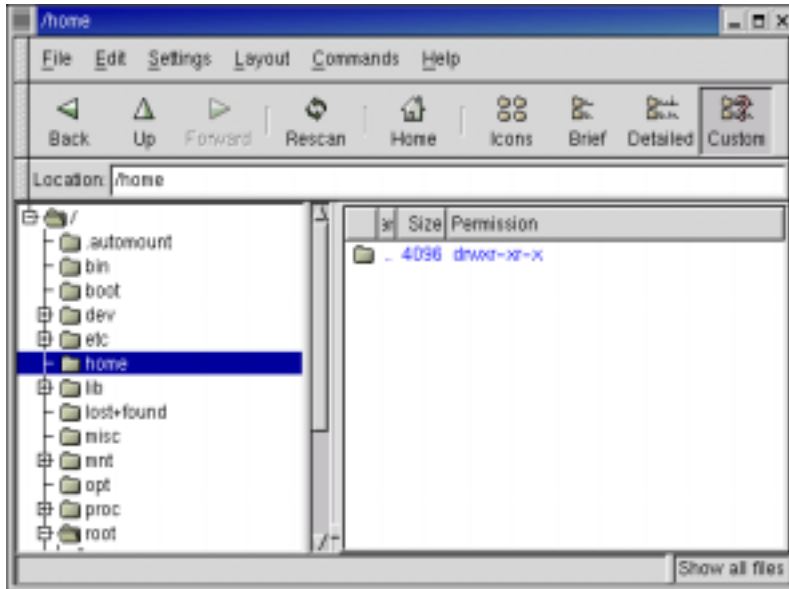



그림 8-13. 주문방식으로 등록부를 현시

중요한 차림표선택항목

GNOME 파일관리기는 몇개의 지령과 선택항목을 제공하는 5 개의 차림표를 가지고 있다. 그러면 파일관리기에서 더 일반적으로 리용된 차림표선택항목을 잠간 보기로 하자. 보다 중요치 않은 다른 선택항목은 GNOME 조종판에서 나타나기는 하지만 이 책에는 주지 않는다. GNOME 파일관리기에 대한 완전한 정보는 기본 GNOME System 차림표에서 Help System 항목을 선택하여 찾을수 있는 GNOME 도움말체계에서 볼수 있다.

File → Create New Window

GNOME 파일관리기는 서로 다른 등록부를 현시하는 여러개의 창문을 연다. 두개의 GNOME 파일관리기창문사이에 그것을 끌어다놓기하여 펼쳐 진 등록부사이에서 파일을 옮길수 있다.

File → Copy

파일을 새 위치에 복사하려면 현시부에서 파일을 선정하고 File → Copy 를 선택해야 한다. Copy 대화칸은 그림 8-14 에 보여 준것과 같다. Browse 단추를 찰칵하여 목적등록부를 선택하거나 본문마당에 직접 목적경로를 써넣으면 된다.

Copy 대화칸을 호출하는 다른 방법은 현시부의 파일을 찰칵할 때 생기는 Pop-up 차림표에서 Copy 를 선택하는것이다.

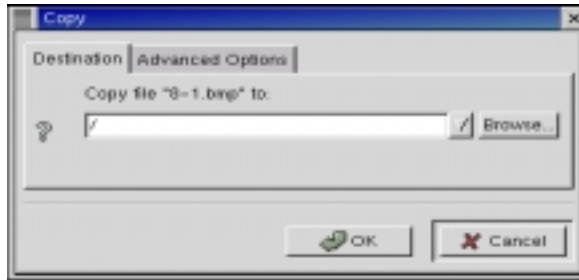



그림 8-14. 파일복사

File → Delete

파일을 삭제하려면 현시부에서 파일을 선정하고 File → Delete 를 선택해야 한다. 그러면 파일을 삭제하거나 취소하겠는가를 문의하는 확인대화칸이 나타난다. 일단 확인하면 이 동작을 복귀할수 없다. 파일을 지우는 다른 방법은 현시부에 있는 파일을 찰칵하여 얻어 지는 pop-up 차림표에서 Delete 를 선택하는것이다.

File → Move

파일을 새 위치로 옮기려면 현시부에서 파일을 선정하고 File → Move 를 선택해야 한다. 그러면 그림 8-14 에 보여 준 Copy 대화칸과 비슷한 Move 대화칸이 나타난다. 여기서 목적지를 정의할수 있다. Move 대화칸을 호출하는 다른 방법은 현시부의 파일을 찰칵하여 얻어 지는 Pop-up 차림표에서 Move 를 선택하는것이다.

Layout → Sort By

현시부에 현시된 파일과 등록부들의 순서를 바꾸려면 Layout → Sort By 를 선택해야 된다. 그러면 그림 8-15 에 보여 준것과 같은 대화칸이 나타난다. 이 대화칸에서 파일과 등록부를 이름, 파일형태, 크기, 호출시간, 변경, 수정시간에 따라 정렬하도록 선택할수 있다. 뿐만아니라 대화칸의 아래부분에 있는 두개의 검사칸을 리용하여 대소문자 구별을 하지 않는다거나 또는 표준정렬순서를 반대로 하도록 설정할수 있다.

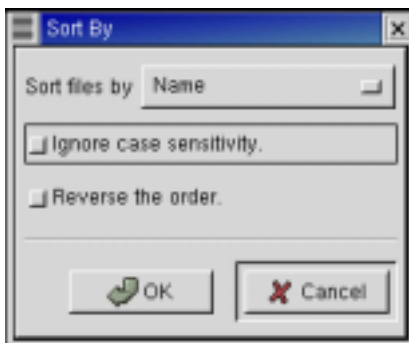



그림 8-15. 정렬순서의 변경

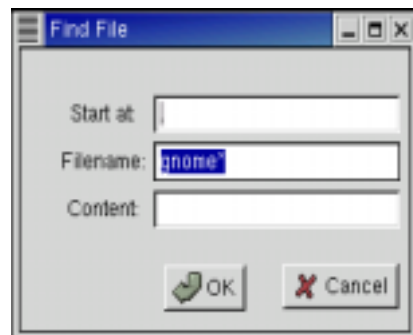



그림 8-16. 파일탐색

Commands → Find File

콤퓨터에서 파일을 찾기 위해서는 GNOME 파일관리기에서 Commands → Find File 선택항목을 리용할수 있다. 그러면 그림 8-16 에 보여 준 Find File 대화칸이 나타난다.

이 대화칸에서는 세개의 파라메터를 정의할수 있다.

- Start At 마당을 리용하여 탐색을 시작할수 있는 장소
- Filename 마당을 리용하여 탐색할수 있는 파일이름
- Content 마당을 리용하여 파일을 탐색할수 있는 본문

파일이름을 정의할 때 통용문자를 리용할수 있다. 례를 들어 gi* 는 gi 로 시작하는 모든 파일들을 조사하며 *gi 는 gi 로 끝나는 모든 파일을 조사한다. 탐색을 시작하는 등록부를 정의할 때 완전한 등록부경로(/home/username 과 같은), Linux 체계의 웃준위 등록부를 나타내는 빗선(/), 현재 선택된 등록부들을 나타내는 점(.)을 지정할수 있다. 찾았으면 탐색결과가 그림 8-17 과 같이 창문에 현시된다.

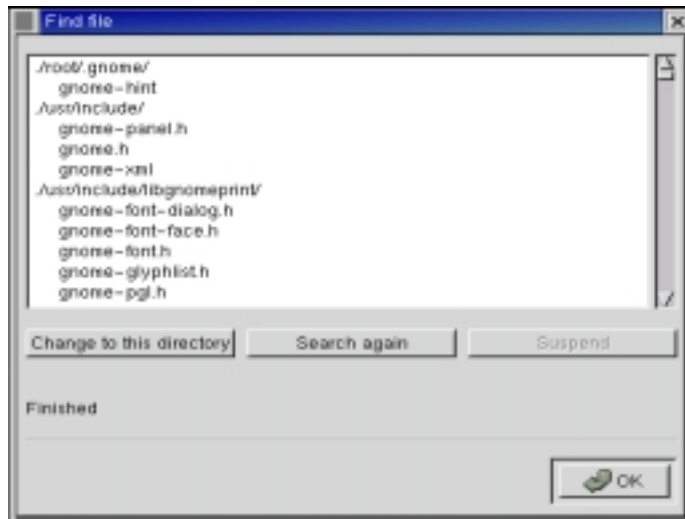



그림 8-17. 파일탐색결과

▶미리 알아두기

지금까지의 과정을 통하여 선택한 X Windows 환경을 아주 안전하게 조종할수 있어야 한다. 따라서 GNOME 의 기초조작 다시말하여 응용프로그람을 기동하고 창문을 관리하는것 등을 할수 있게 되였다.

제 9 장 《GNOME 과 X Windows 에서 응용프로그람의 리용》에서는 GNOME 에서 실제적인 응용프로그람을 리용하는 방법을 고찰한다. 이 응용프로그람은 gedit 와 같은 간단한 응용프로그람, GNOME 문서편집기로부터 말단창문과 같은 편의프로그람, 전문영상편집프로그람제품인 The GIMP 와 같은 고급한 쏘프트웨어에 이르기까지의 모든것을 포괄한다.

제 9 장. GNOME 과 X Windows 에 의한 응용프로그람의 리용

이 장에서는 일부 X Windows 와 GNOME 응용프로그람과 GNOME 탁상에서 그것들이 어떻게 실행되고 리용되는가를 고찰한다. 이러한 응용프로그람에는 xterm, GNOME 파일관리기(Midnight Commander 라고 하는), The GIMP 로 알려 진 도형처리프로그람과 X Windows 편의프로그람이 있다.

이 프로그람은 두개의 기준가운데서 하나에 기초하여 선택되였다. 즉 그것들은 Linux 조작에서 항상 쓸모 있거나 많은 Linux 응용프로그람의 능력범위를 나타낸다. 이 장에서는 기타 X Windows 응용프로그람을 실험해 볼수 있는 확신성을 준다.

xterm 의 리용

X Windows 가 리용하기 좋고 Linux 사용자에게 친근감을 주며 완전한 도형처리응용프로그람을 작성할수 있게 하지만 실제로 Linux 는 Unix 클론과 같이 지령행에 중점을 둔 조작체계이다.

지령재촉문을 리용하지 않고도 Linux 를 매일 쓸수 있지만 Unix 조작체계와 같은 실제적인 능력과 유연성을 리용하려면 적어도 지령행의 세계에로 넘어 가야 한다.

다행히도 이것은 안전한 X Windows 를 버린다는것을 의미하지 않는다. xterm 프로그람은 X Windows 환경에 완전히 통합된 지령재촉문창문을 제공하는데 이 창문은 이동띠(scroll bar), 크기를 바꿀수 있는 창문 그리고 기타 X Windows 응용프로그람을 복사하고 붙이는 능력 즉 표준 Linux 조종탁에 없는 모든 특성들을 포함하고 있다.

주의	Programs 차림표의 System 보조차림표에서 Regular XTerm 을 선택하여 Red Hat Linux 7.1 의 **xterm** 창문을 기동할수 있다. GNOME 에는 GNOME 말단으로 알려 진 **xterm** 의 조종탁이 있다. 그러나 KDE 를 비롯하여 다른 Linux 도형처리환경에서도 **xterm** 을 리용할수 있기때문에 이 장에서는 그것을 상세하게 고찰한다. 대부분의 동일한 지령은 GNOME 말단에서 작업한다. 그리고 일부는 큰 풀이표(extra dash)와 같은 약간한 변경을 요구한다. 이 장에서는 두가지 형태의 말단사이 중요한 기능상 차이만을 고찰한다.

추가 xterm 창문을 기동하려면 다음지령을 주어서 다른 xterm 창문의 지령에서 기동할수 있다.

```
$ xterm &
```

또는 어떤 리유로 그 경로에 xterm 이 없으면 다음지령을 줄수 있다.

```
$ /usr/X11R6/bin/xterm &
```

일러두기	대부분 파일들의 위치는 **locate** 지령으로 알아 낼수 있다. **xterm** 이 어디에

위치하고 있는가를 보자면 **locate xterm** 지령을 실행해야 한다. 그런데 **locate** 지령은 새롭게 작성되거나 움직인 파일들에 대해서는 실행되지 않는다. 이 경우에는 제 13 장 《Linux 지령의 기본개념》에서 보게 되는 **find** 지령을 리용해야 한다.

대신에 다음의 지령을 리용하여 다른 xterm 창문의 지령행으로 추가 GNOME 말단창문을 기동할수 있다.

$ **gnome-terminal &**

혹은 표준 GNOME 조종판에서 말단아이콘을 찰칵할수 있다.

기정으로 이 지령은 너비가 80 문자이고 길이가 24 행인 이동띠를 가진 창문을 연다. 창문은 정보열람과 작업을 위하여 크기를 조절할수 있다. 그림 9-1 에 보여 준바와 같이 Red Hat 7.1 에서 새 xterm 창문의 기정색방안은 백색배경에 검은색문자이다. 창문의 기정제목은 현재 사용자와 등록부를 정의한다.

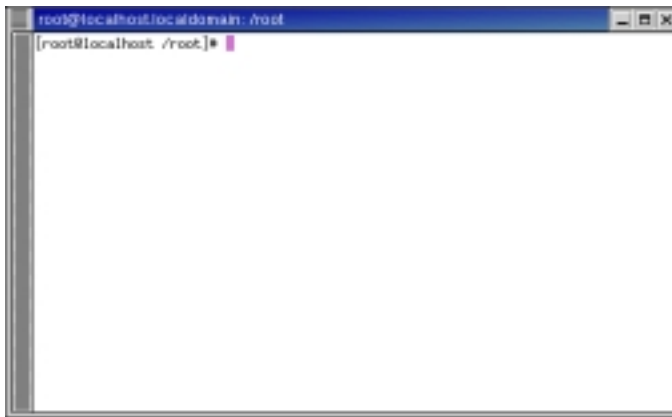



그림 9-1. xterm 창문

xterm 의 전용화

xterm 프로그람은 높은 속도로 전용화할수 있다. 그가운데서 다음의 특성을 변경시킬수 있다.

- 창문의 서체
- 색방안
- 기정창문크기
- 화면에 새 창문의 배치

이 절에서는 이 선택항목과 새 xterm 창문을 기동할 때 그것을 리용하는 방법을 고찰한다.

처음에 이 유연성은 실제값을 가지지 않은것처럼 보이지만 실제로는 값을 가진다. 서로 다른 현시장치에서 각이한 색조합은 가장 보기 좋은 본문을 만들므로 색방안의 변경은 중요한 특징으로 된다.

기발과 파라메터들 즉 Unix 속성안내서

앞으로 나가기전에 Unix 지령의 실행방법에 대한 속성프로그람을 알아야 한다. Unix 에 대한 완전한 소개와 Linux 에서 리용하는것과 같은 Unix 지령은 제 13 장과 제 14 장에서 고찰한다.
프로그람(혹은 지령)을 실행하려면 프로그람의 이름(그리고 될수록 프로그람의 완전한 위치)을 간단히 입력하고 Enter 건을 눌러야 한다.
례를 들어 xterm 을 기동하려면 다음의 지령을 주어야 한다.

```
$ xterm
```

또는 xterm 이 있는 현행등록부가 경로에 없으면 다음의 지령을 주어야 한다.

```
$ /usr/X11R6/bin/xterm
```

이 경로는 Linux 가 지령을 탐색하는 장소이다. 다음의 지령으로 경로를 탐색할수 있다.

```
$ echo $ PATH
```

대부분의 프로그람은 자기가 동작하는 통로를 바꾸거나 처리되는 정보를 제공하는 정보를 요구할수도 있고 받아 들일수 있다. 두가지 형태의 정보 즉 기발과 인수를 지령으로 제공할수 있다.
인수는 처리되는 프로그람에 제공되는 정보이다. 그것은 파일이름으로부터 본문에 이르기까지의 탐색하는 모든것이 될수 있다.
기발은 프로그람의 동작을 바꾸는 선택항목이다. 기발은 지령뒤에 나타나며 하나이상의 공백으로 구분되고 앞에는 풀이표가 있다. 례를 들어 기발 –help 는 xterm 이 도움말통보를 인쇄하게 한다. 즉

```
$ xterm –help
usage:

        xterm –선택항목 … -e command args
```

여기서 선택항목들에는 다음과 같은것이 포함된다.

-help	이 통보문을 인쇄하여 배포한다.
-display display_name	련결하려는 X 봉사기
-geometry geom	크기(문자단위)와 위치
-/+rv	비데오를 절환
-bg color	배경색
-fg color	전경색-bd color 경계색
-bw number	화소단위의 경계너비
-fn fontname	표준본문서체
-iconic	최소화시작
-name string	의뢰기실체, 아이콘, 제목문자렬
-title string	제목문자렬
-xrm resourcestring	추가적인 자원을 정의
-/+132	행스위치를 절환
-/+ah	on/off 밝기절환

-/+ai 능동아이콘을 절환

복합기발은 기발이 정확한 기능을 요구하는 정보를 제공하기 위하여 작성될수 있다. 례를 들어 **xterm** 색을 설정하기 위하여 **–fg** 와 **–bg** 기발을 리용한다. 지령

$ xterm –fg white –bg black

은 검은색배경에 흰색문자렬을 가진 창문을 만든다. 한편 한가지 간단한 기발은 글자의 이름(&)이다. 실례로 지령

$ xterm &

은 다른 **xterm** 창문을 열고 현재의 **xterm** 창문을 지령행대면부로 복귀시킨다.
한편 인수들은 구성선택항목은 아니지만 처리될 프로그람에 제공되는 정보이다. 인수에는 풀이표(-)가 붙지 않는다. 인수들은 흔히 열리고 처리되거나 편집되는 파일들의 이름들이다. 즉 다음의 실례에서와 같이 인수는 파일 testfile 을 편집기 **emacs** 로 편집하기 위해서 열려야 한다는것을 나타낸다.

emacs testfile

색방안의 설정

다양한 기발을 리용하여 경계색으로부터 화살표의 색에 이르기까지 xterm 창문의 색방안을 모두 설정할수 있다. 이 절에서는 색방안의 기본요소 즉 전경색, 배경색, 유표색을 고찰한다.

X Windows 색의 해석

xterm 창문의 색설정에 리용되는 기발을 보기전에 X Windows 에서 색갈을 어떻게 구성하는가를 알아야 한다.

X Windows 에서 색은 RGB(붉은색-풀색-푸른색)형태로 정의된다. RGB 형태는 매수가 0~255 의 값을 가지는 세개의 수로 이루어 진다. 수는 붉은색, 풀색, 푸른색통로의 값을 표시하는데 여기서 0 은 색갈이 없다는것을, 255 는 가능한 색의 가장 큰 값을 의미한다.

다행히도 설계된 색을 얻기 위하여 색통로조합을 밝히지 않아도 된다. 작업은 이미 수행되였으며 X Windows 가 색이름을 수로 표시한 조에 표시하는데 리용하는 파일에 기록된다. 표준으로 rgb.txt 라고 하는 이 파일은 일반적으로 /usr/X11R6/lib/X11/에서 보게 된다.

이 절에서 작업한것과 같은 색에 대한 고찰을 쉽게 하기 위해서 아래에 Red Hat Linux 7.1 rgb.txt 파일에서 선택된 값목록을 주었다.

```
snow            GhostWhite      PapayaWhip      LemonChiffon
AliceBlue       LavenderBlush   MistyRose       white
DarkSlateGray   DimGray         gray            MidnightBlue
NavyBlue        SlateBlue       blue            SteelBlue
turquoise       cyan            DarkGreen       SeaGreen
LawnGreen       green           GreenYellow     DarkKhaki
```

LightYellow	yellow	gold	osyBrown
IndianRed	sienna	beige	wheat
tan	chocolate	DarkSalmon	orange
tomato	red	DeepPink	pink
maroon	magenta	violet	orchid
purple	DarkCyan	DarkRed	LightGreen

주의 비데오현시장치의 색계조와 실행하고 있는 응용프로그람들에 기초하면 리용하는 색들이 기대하는 방식을 전혀 나타내지 않는것을 보게 된다. 이것은 비데오카드가 제한되고 체계마다 다르기때문이다.

배경과 전경색

전경과 배경색을 설정하여 화면과 문자렬을 조종할수 있다. 본문은 창문배경에 할당된 배경색에 대하여 전경색으로 나타난다.

xterm 창문의 기정색은 백색배경에 검은색본문이다. 전경과 배경색을 설정하려면 –fg 와 –bg 기발을 리용해야 한다.

$ xterm –fg white –bg black &

이 지령은 표준 xterm 창문에 상반되는 색방안을 정확히 제공하며 배경을 검은색으로, 전경을 흰색으로 설정하게 한다. 이 결과는 그림 9-2에서 본것과 같다.

주의 **-bg** 와 **–fg** 기발은 xterm 에 대하여 정의하지 못한다. 실제로 그것들은 대부분의 X Windows 응용프로그람의 색방안을 변경시키는데 리용될수 있는 표준 X windows 기발들이다. 그러나 이 기발들은 **gnome-terminal** 지령에서 동작하지 못한다. GNOME 말단을 위한 정확한 기발을 찾으려면 **gnome-terminal-help** 지령을 실행해야 한다.

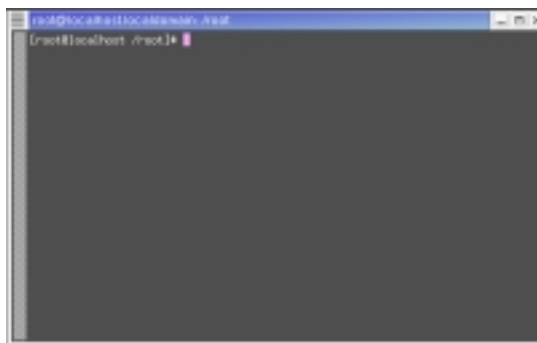



그림 9-2. 색방안의 조종

유표색

색을 설정하는 마지막실례를 실행시켰을 때 유표는 표준 xterm 색방안과 같은 색으로 남아 있다. -cr 기발을 리용하면 이 색을 변경시킬수 있다. 례를 들어 검푸른색의 유표를 원한다면 다음의 지령을 리용하여야 한다.

$ Xterm – bg white – fg black – cr MidnightBlue &

창문의 크기와 위치설정

xterm 창문(혹은 다른 임의의 프로그람창문)을 열면 창문의 크기는 표준으로 설정되며 창문의 위치는 창문관리기를 구성하여 결정된다(제 12 장 《고급한 X Windows 구성》에서 주요파일의 xterm 지령을 보게 된다.).

그러나 –geometry 기발을 리용하여 X Windows 에서 창문의 크기와 위치를 정의할 수 있다. 이 기발은 표준 X Windows 기발이며(-bg 와 –fg 기발과 같이) 대부분의 X Windows 프로그람과 함께 작업한다.

기발의 기본문법은 –geometry 너비*높이+x+y 이다. 이 지령의 인수를 아래에 주었다.

- 너비는 창문너비를 화소단위로 정의한다.
- 높이는 창문높이를 화소단위로 정의한다.
- x 는 현시장치의 왼쪽면으로부터 창문의 수평위치편차를 화소단위로 정의한다.
- y 는 현시장치의 웃면에서부터 창문의 수직위치편차를 화소단위로 정의한다.

Xterm 은 –geometry 기발을 리용할수 있지만 특징적이지 못한 비표준방법으로 정보를 조종한다. 이에 대하여 생각할 때 말단창문의 높이와 너비를 화소단위로 정의하면 조금도 느끼지 못한다. 대신에 xterm 은 높이와 너비를 리용하는 서체에 따라 창문을 현시하고 조절하는 렬의 총수와 문자들의 행으로 해석한다.

표준창문의 크기는 문자단위로 24 행에 80 렬이다. 그러나 작은 문자단위의 10 렬× 10 행창문을 왼쪽면에서 200 화소단위, 화면의 웃면에서 300 화소단위 떨어 진 곳에 배치하려면 어떻게 하겠는가? 그림 9-3 과 같은 창문을 얻기 위해서 다음의 지령을 리용할수 있다.

$ xterm –geometry 10×10+200+300 &

창문의 위치는 화면의 왼쪽 모서리와 웃모서리에 관계된다는것을 명심하여야 한다.s

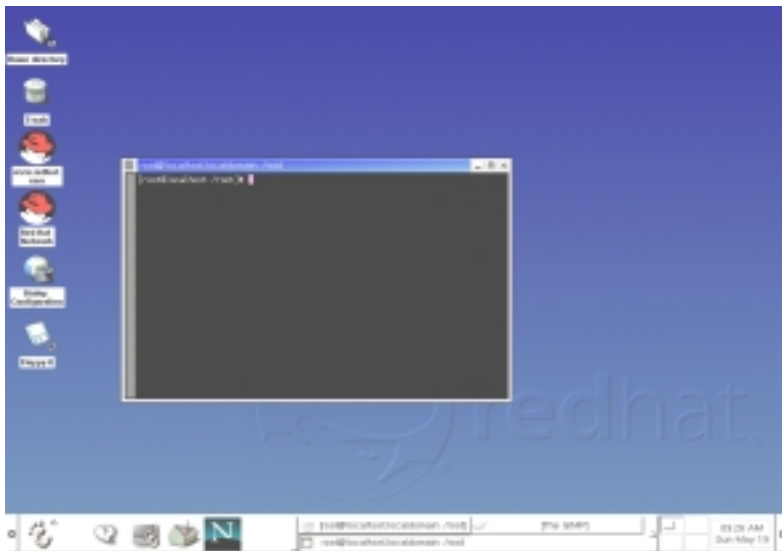



그림 9-3. –geometry 기발의 리용

주의	GNOME 말단에 부합되는 지령은 **gnome-terminal-geometry=10x10+200+300 &**이다. 그러나 이 지령의 효과는 GNOME 말단들이 최소한 25 렬의 너비를 가지기때문에 제한성을 가진다.

-fg 와 –bg 기발을 추가하여 창문이 열릴 때 그것의 크기를 정하는것과 동시에 표준적인 색방안을 변경시킬수 있다.

$ **xterm-geometry 10*10+200+300 –fg DarkRed -bg cyan &**

그 결과는 그림 9-4 와 같다.

그림 9-4. –geometry 기발과 color 기발의 결합

xterm 창문에서 응용프로그람의 실행

xterm 창문의 목적이 지령을 실행하는 지령행을 리용하는것이라면 xterm 창문을 기동할 때 기정으로 실행되는 일정한 문자방식프로그람들을 요구할수 있다. 례를 들어 작업과정에 xterm 창문에서 top 를 계속 실행하려면 xterm 을 펼칠 때 top 도 기동하도록 해야 한다. 이것은 –e 기발을 리용하여 수행할수 있다. 즉

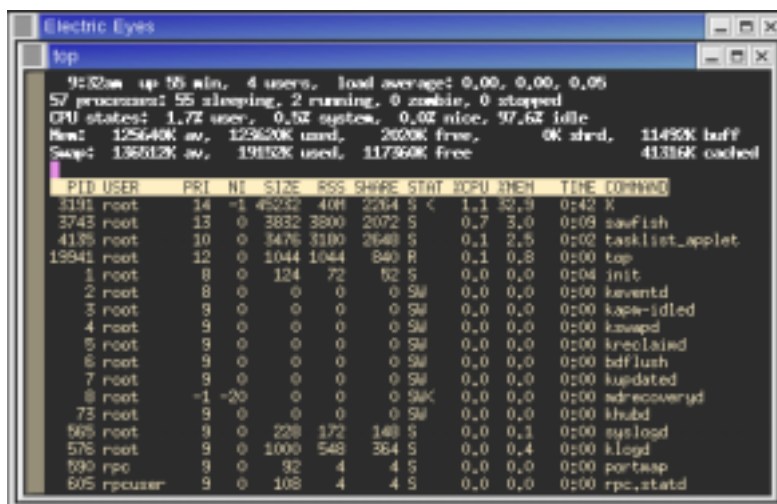
$ **xterm –e top &**

이 지령을 실행하면 그림 9-5 와 같은 창문이 나타난다.

창문에서 프로그람실행이 끝나면 xterm 창문은 닫긴다.

–e 기발을 리용할 때 xterm 에서 기정창문제목이 –e 기발로 기동된 프로그람의 이름으로 된다는것을 명심해야 한다.

주의	**-e** 기발이 **gnome-terminal** 지령과 함께 리용될 때에는 기정창문제목을 설정하지 못한다.

그림 9-5. xterm 창문이 펼쳐 질 때 top 의 기동

서체의 설정

xterm 에서 리용되는 기정서체는 보통 빨리 읽을수 있으며 일반수준의 현시장치에 적합하다. 그러나 특별히 크거나 작은 현시장치를 리용하거나 례외적으로 낮거나 높은 환경을 리용한다면 xterm 창문에서 리용하는 서체를 변경시켜야 한다.

-fn 기발을 리용해서 이것을 실현할수 있다. 그러면 이 서체를 어떻게 결정하겠는가? 서체는 간단히 7 × 13 으로 쓸수 있는데 이것은 7 화소단위의 너비와 13 화소단위의 높이를 가리킨다. 또는 24 점라틴문자묶음, 중간높이, Sony 서체생산지에서 fixed 라고 이름을 단 Roman-style 서체를 가리키는 –sony-fixed-medium-r-normal-24-230-75-75-c-120-iso8859-1 과 같이 복잡한 이름으로 해석될수도 있다..

X Windows 는 X Windows 체계에서 리용할수 있는 서체목록을 주는 xlsfonts 라는 프로그람을 제공한다. 체계에서 리용할수 있는 서체목록을 보기 위하여 xterm 창문의 다음지령을 리용한다.

```
$ xlsfonts | more
```

지령의 |more 부분은 앞페지로 가기 위해서 공백건을 눌러 서체의 긴 목록을 훑어보게 한다.

그러므로 10 × 20 서체를 선택하여 그림 9-6 에 보여 준것과 같은 큰 서체를 가진 창문을 얻기 위해서는 다음의 지령을 리용해야 한다.

```
$ xterm –fn 10x20 &
```

이것을 그림 9-1 과 비교할 때 서체만 커진것이 아니라 xterm 창문도 역시 같은 수의 행과 렬을 가진 더 큰 서체에 대응하게 더 커진다는것을 명심해야 한다.

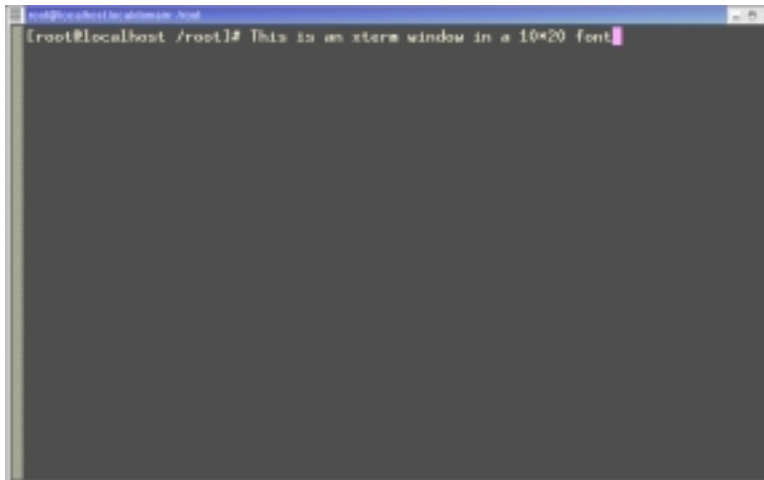



그림 9-6. 10 × 20 서체

리론적으로 서체는 –fn 기발 그리고 xterm 지령과 함께 리용할수 있다. 그러나 실제로는 이렇게 작업하지 않는다. xterm 의 특징은 그것이 fixed-width 서체 즉 모든 문자가 그것의 상대적인 크기에 관계없이 꼭 같은 공간을 차지하는 서체를 요구한다는것이다.

Microsoft Windows 의 세계에서 fixed-width 서체의 실례는 Courier 이다.

대조적으로 많은 서체들은 비례적으로 공간을 차지한다. 이것은 문자 《i》가 《w》보다 더 좁은 문자이기때문에 《w》보다 더 작은 공간을 차지한다는것을 의미한다. xterm 은 그림 9-7 에서 볼수 있는바와 같이 이 비례적인 공간을 차지하는 서체를 조종하는 좋은 일감을 수행하지 못하는데 여기서 서체는 Times 족에서 선택한것이다.

```
mj@testlinux:/home/mj
ftpconversions      mime.types              securetty
ftpgroups           minicom.users           security
ftphosts            modules.conf            sendmail.cf
gimp                modules.conf.bak        services
gpm-root.conf       modules.conf.org        shadow
group               modules.conf.org.old.1  shadow-
gshadow             motd                    smrsh
gshadows-           mtab                    snmp
gtk                 mtools.conf             sound
host.conf           named.boot              ssh
hosts               named.conf              sysconfig
hosts.allow         news                    sysctl.conf
hosts.deny          nmh                     syslog.conf
httpd               nsswitch.conf           termcap
indentd.conf        openldap                updatedb.conf
im_palette-small.pal opt                    vmware
im_palette-tiny.pal pam.d                   wvdial.conf
im_palette.pal      paper.config            xinetd.conf
imrc                passwd                  xinetd.d
info-dir            passwd-                 yp.conf
init.d              passwd.OLD              ypserv.conf
[mj@testlinux mj]$
```

그림 9-7. 비례하여 공간을 차지하는 서체의 리용

아래에 렬거된것들은 Red Hat Linux 7.1 에 포함된 X Windows 배포물에서 리용할수 있는 일반적인 고정너비서체들이다.

5 × 7	6 × 13bold	7 × 14bold	9 × 15bold
5 × 8	6 × 9	8 × 13	10 × 20
6 × 10	7 × 13	8 × 13bold	12 × 24
6 × 12	7 × 13bold	8 × 16	
6 × 13	7 × 14	9 × 15	

다른 서체를 시험해 보려면 서체목록을 보기 위하여 xlsfonts 지령을 리용하고 서체를 선택한 다음 창문에 서체를 현시하기 위하여 xfd 지령을 준다. 례를 들어 이미 본 Sony 서체 –sony-fixed-medium-r-normal-24-230-75-75-c-120-iso8859-1 을 현시하려면 다음의 지령을 리용해야 한다.

$ xfd –fn -sony-fixed-medium-r-normal-24-230-75-75-c-120-iso8859-1

서체이름을 정의하려면 –fn 기발을 리용해야 한다는것을 명심해야 한다. 서체는 그림 9-8 에 보여 준것처럼 창문에 현시된다. 창문을 닫기 위하여 Quit 단추를 찰칵한다.

그림 9-8. xfd 지령으로 서체를 현시

기타 특성

xterm 은 여기서 서술한것보다 더 많은 특성들을 가지고 있다. 이 자료는 xterm 사용설명서페지에서 볼수 있다. 사용설명서(manual 의 략자)페지는 정의한 프로그람을 리용하는 방법에 대한 정보를 제공한다. xterm 사용설명서페지를 호출하기 위하여서는 xterm 창문에 있는 다음의 지령을 리용해야 한다.

```
$ man xterm
```

이 지령을 실행하면 그림 9-9 와 같은 사용설명서페지가 현시된다. 사용설명서페지를 흐르게 하기 위하여 화살이 있는 건을 리용한다. 사용설명서페지에서 탈퇴하기 위해서는 q 를 입력해야 한다.

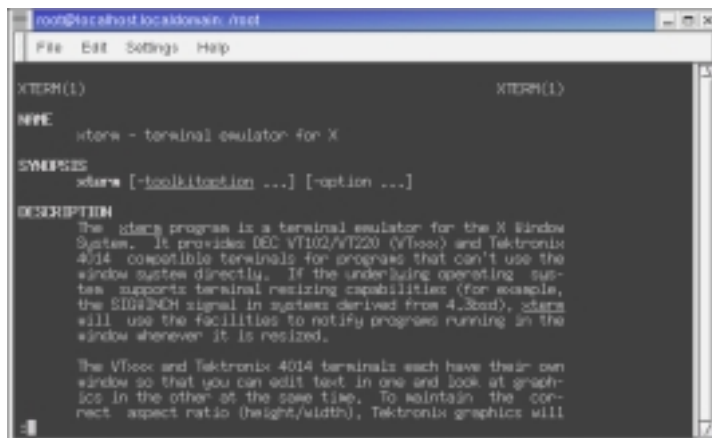



그림 9-9. xterm 사용설명서페지

주의 이 책을 쓸 당시 GNOME 말단에서 리용할수 있는 사용설명서페지는 없었다. 이 프로그람에 대한 더 상세한 정보를 보기 위해서는 GNOME 말단창문을 펼치고 Help→GNOME Terminal Users Guide를 선택해야 한다.

X Windows 응용프로그람

지금까지는 X Windows 응용프로그람의 동작에서 가장 기초적인 두개의 프로그람 xterm과 GNOME 파일관리기를 보았다. 만일 이것밖에 없다면 X Windows가 얼마간의 응용프로그람밖에는 가지고 있지 않고 Windows와 Mac OS도 작용이 있는 곳이라고 불평하는 사람들의 견해가 많아 질수 있다.

그러나 실제로 X Windows 환경에는 강력하고 효률적인 프로그람들이 많이 존재한다. 지어 무료도구가운데는 업무용응용프로그람의 특성과 비슷한것들이 있다.

이 절에서는 GNOME Office Suite의 두 프로그람 즉 The GIMP 도형처리편의프로그람과 Gnumeric 표처리프로그람을 고찰한다. 이 프로그람들은 많은 능력을 가지고 있기때문에 이 절에서는 프로그람에 대한 개괄과 프로그람에서 리용가능한 완전한 정보들만 본다. The GIMP나 Gnumeric와 같은 프로그람에 정통하자면 매 프로그람마다 책 한권씩은 있어야 한다.

주의 GNOME Office Suite의 다른 특성들에는 AbiWord라는 문서편집프로그람, Visio 형도형들과 흐름도들을 위한 Dia 프로그람, GNOME-PIM 개인정보관리기가 있다.

The GIMP

The GIMP(GNU Image Manipulation Program)는 무료로 리용할수 있는 Linux 프로그람이며 Adobe Photoshop나 Corel PhotoPaint와 같은것들에 대응하는 완전히 특색 있는 영상편집도구이다. 그림 9-10에는 실행하여 여러가지 대화칸과 도구를 펼쳐놓은 The GIMP를 보여 준다. The GIMP는 GNOME에 든든하게 통합시키기 위하여 설계되였으며 GNOME에 대한 표준적인 영상편집프로그람과 아주 강력한 GNOME 응용프로그람의 수단으로 간주된다. GNOME은 Red Hat Linux 7.1에 한 부분으로 포함되여 있다.

General Public License에서 제공된 The GIMP는 켈리포니아-버클리종합대학의 스펜서킴볼과 페터매티즈에 의하여 부단히 개발되고 있다.

The GIMP는 많은 붓종류, 특수한 효과를 가진 려파기, 정보처리기능을 가진 가위, Bezier 선택령역, 층, 투명도와 24bit 영상지원기능을 비롯하여 완전히 특색 있는 영상편집프로그람묶음들로 기대되는 모든 도구들을 포괄하고 있다.

The GIMP는 또한 꽂개식의 조작기구도 제공한다. 많은 꽂개식도구들은 인터네트상에서 무료로 리용할수 있는데 얼룩무늬만들기, 구넘기기, 쪽무늬만들기, 렌즈불빛만들기 등을 비롯하여 다양한 과제들을 수행한다.

그림 9-10. The GIMP

이외에 Red Hat 7.1 에 포함된 판본(The GIMP)에는 무료쏘프트웨어에 흔히 없는 상세한 지령과 문서를 제공하는 내리적재할수 있는 설명서들이 있다. 이것은 manual gimp.org 에 있는 The GIM Web 싸이트에서 모두 리용할수 있다.

다음의 지령으로 xterm 창문에서 The GIMP 를 실행할수 있다

```
$ gimp &
```

또는

```
$ /usr/bin/gimp &
```

대신에 The GIMP 를 기동하기 위하여 기본 GNOME 체계차림표에서 Programs → Graphics → The GIMP 를 선택한다.

The GIMP 는 그림 9-10 에서 보여 준 도구띠와 많은 기타 설계화면들을 현시하는것으로부터 시작한다. 이 절이 목적하는데서 가장 중요한 창문은 도구띠이다. 그림 9-11 에 보여 준 이 창문은 기본 File 차림표를 포함한 넓은 범위의 도구들에 대한 호출을 제공한다.

The GIMP 의 몇가지 기본과제 즉 영상펼치기와 보관, 자르기, 창문을 영상파일로 표시하기에 대해 간단히 고찰하자. The GIMP 특성들을 완전히 배우기 위해서는 앞에서 언급된 The GIMP 의 Web 싸이트에 있는 쏘프트웨어에 대한 설명서를 참고해야 한다.

그림 9-11. The GIMP 도구띠

영상열기

영상을 열기 위해서 File → Open 을 선택한다. 그러면 그림 9-12 에 보여 준것과 같은 대화칸이 현시된다.

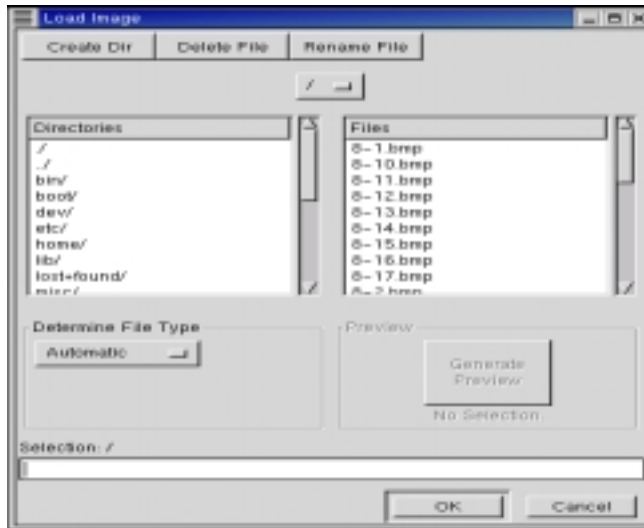



그림 9-12. The GIMP Load Image 창문

이 대화창문은 Mac 와 같은 방식으로 동작한다. 파일들의 목록에서 현행등록부의 이름을 볼수 있다. 자기 어미등록부의 목록을 얻기 위하여 등록부이름을 찰칵할수 있고 이것들가운데서 하나를 선택하여 그 등록부를 빨리 변경할수 있다.

요구하는 파일이름을 선택한다. 어느 파일이 필요한가 하는것이 명백치 않으면 Load Image 창문의 Preview 부분에서 Generate Preview 단추를 찰칵하여 개요를 줄수 있다. 필요한 파일이 보이면 Ok 단추를 찰칵한다. 파일은 미리보기창문에 나타나고 대화칸은 닫긴다.

영상의 보관

영상을 보관하려면 The GIMP 기본차림표를 리용해야 한다. 능동영상에서 작업하는 과정에 영상의 임의의 곳을 찰칵하면 그림 9-13 에서 보여 준것과 같이 The GIMP 의 완전한 차림표가 나타난다.

그림 9-13 .The GIMP 기본차림표

Save Image 대화칸을 열려면 The GIMP 기본차림표로부터 File → Save 를 선택해야 한다. 그림 9-14 에 보여 준 결과는 Load Image 대화칸과 비슷하다. 등록부조종은 같은 방식으로 진행한다. 그리고 파일목록의 파일이름을 찰칵하여 이미 존재하는 파일을 선택한다. OK 단추를 찰칵하여 현재파일을 Selection 마당에 현시된 이름으로 보관한다.

영상을 새로운 파일이름으로 보관하려면 Selection 마당에 간단히 요구하는 이름을 입력해야 한다.

영상을 보관하려면 영상파일의 형태를 정의해야 한다. 대화칸의 중간에는 Determine File Type 단추가 있다. 이 단추를 찰칵하여 리용할수 있는 파일형태목록을 볼수 있다. 목록에서 파일형태를 선택하고 준비되면 OK 를 찰칵한다.

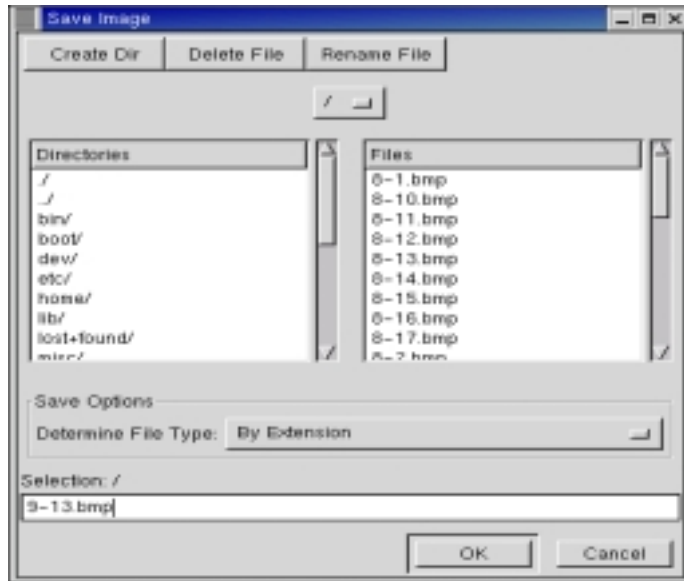



그림 9-14. The GIMP 영상보관대화칸

영상의 자르기

영상자르기는 간단하다. 즉 자르기로 절환하고 자르려고 하는 구역을 선택한 다음 Crop & Resize Information 창문에서 Crop 단추를 누른다.

자르기로 절환하려면 영상을 찰칵해야 한다. 펼쳐 진 차림표에서 Tools → Transform Tools → Crop & Resize 를 선택한다. 자르려는 구역을 선택하려면 왼쪽마우스단추로 영상에 4 각형을 끌어 낸다. 그림 9-15 는 사용자가 마우스끌기한 곳을 나타내는 4 각형과 함께 보임창문에 현시된 영상을 보여 준다.

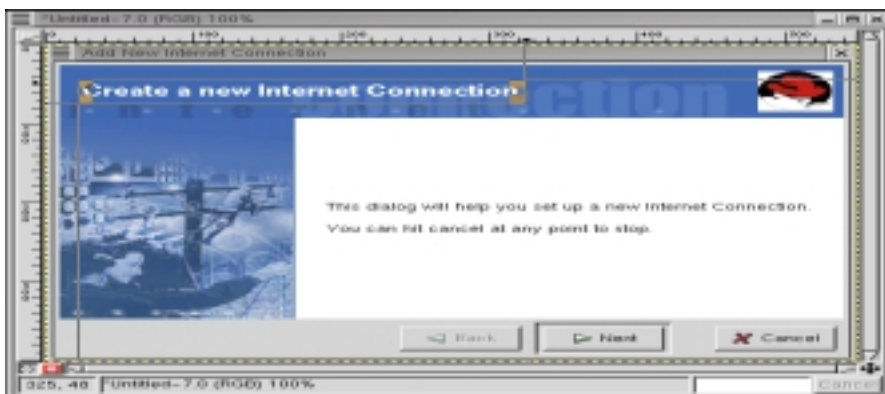



그림 9-15. 자르기구역선택

일단 자르려는 구역이 선택되였으면 간단히 Crop & Resize Information 창문에서 Crop 단추를 찰칵한다. 우의 실례에서 결과는 그림 9-16 과 같아야 한다.

Create a new Internet Connection

그림 9-16. 자르기결과

자르기한 결과가 마음에 들지 않으면 영상을 오른쪽찰칵한다. 초기영상으로 돌아 가려면 펼쳐 진 차림표에서 File→Revert 를 선택해야 한다.

창문의 잡아넣기

The GIMP 에 대한 일반적인 리용방법의 하나는 화면시점(shots)을 작성하는것이다. 영상파일에서 열린창문을 넣으려면 Acquire Screen Shot 특성을 리용하여 이것을 할수 있다. 넣으려는 창문이 있으면 File → Acquire → Screen Shot 를 선택하는데 이것은 그림 9-17 에서와 같은 Screen Shot 대화칸을 연다.

그림 9-17. Screen Shot 창문

개별적인 창문이나 탁상의 전체 내용을 얻을수 있다. 개별적인 창문을 얻기 위하여 Single Windows 를 선택한다. 먼저 OK 를 찰칵하고 선택한 창문을 찰칵한다.

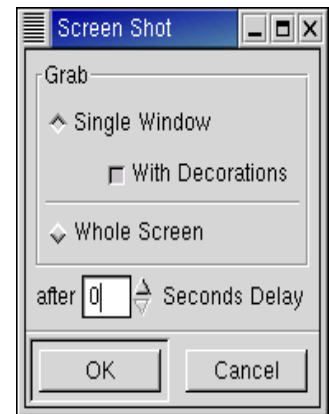
창문의 틀이나 제목띠가 필요 없으면 With Decorations 를 설정하지 않아도 된다. 례를 들어 top 편의프로그람도구의 xterm 창문을 얻으면 전경색과 배경색이 바뀌 지는데 결과는 그림 9-18 에 보여 준것과 같이 전혀 X Windows 를 리용하고 있지 않은것처럼 보여야 한다.

PID	USER	PRI	NI	SIZE	RES	SHARE	STAT	%CPU	%MEM	TIME	COMMAND
1603	root	20	0	10188	9620	2284	R	67.4	20.8	146:51	X
2831	mj	9	0	2736	2736	2228	S	12.3	5.9	0:01	screenshot
2608	mj	9	0	3048	2996	1672	S	5.4	6.4	0:29	sawfish
2830	mj	19	0	1016	1016	812	R	3.3	2.2	0:01	top
2653	mj	9	0	2896	2640	2172	S	2.8	5.7	0:17	deskguide_apple
2651	mj	9	0	3028	2772	2222	S	2.6	6.0	0:18	tasklist_applet
2634	mj	9	0	4404	4032	3140	S	1.6	8.7	0:23	panel
2828	mj	9	0	2496	2496	1864	S	1.3	5.4	0:01	xterm
6	root	9	0	0	0	0	SW	0.9	0.0	0:09	bdflush
2604	mj	9	0	1480	1480	1332	S	0.5	3.0	0:07	gnome-smproxy
2641	mj	9	0	4908	4908	4364	S	0.3	9.4	0:10	gmc
2622	mj	9	0	2080	2080	1676	S	0.3	3.6	0:10	magicdev
2639	mj	9	0	1372	1372	1224	S	0.1	2.6	0:08	xscreensaver
4	root	9	0	0	0	0	SW	0.0	0.0	0:26	kswapd
1	root	8	0	120	120	76	S	0.0	0.1	0:05	init
2	root	9	0	0	0	0	SW	0.0	0.0	0:00	keventd
3	root	9	0	0	0	0	SW	0.0	0.0	0:26	kapm-idled

그림 9-18. top 의 화면시점

대신에 Whole Screen 을 선택하여 Screen Shot 창문에 있는 탁상의 전체 내용을 얻을수 있다. The GIMP 가 탁상에 대한 화면시점을 만들기전에 다른 창문을 닫거나 여는데 필요한 시간을 주기 위하여 Seconds Delay 칸을 설정한다.

일단 창문에서 탈퇴했으면 그 창문의 영상은 관찰자창문에 나타나며 The GIMP 의 다른 영상과 같이 그것을 보관할수 있다.

Gnumeric

Gnumeric 는 작업판의 수값자료를 관리하기 위하여 설계된 GNOME 탁상환경의 또 다른 구성요소이다. 이 프로그람은 Linux 사용자들에게 Microsoft Excel 과 같은 다른 표처리프로그람과 자료를 교환하고 무료로 리용할수 있으며 완전히 특색 있는 표처리프로그람을 제공한다. 그것은 Red Hat Linux 7.1 에 기정으로 설치된다.

일반공개사용허가하에서 제공된 Gnumeric 는 아주 유명한 Miguel de Icaza 와 Jody Goldberg 와 같은 수많은 독자적인 개발자들에 의하여 일정하게 개발되였다. Red Hat Linux 7.1 에 들어 있는 판본은 0.61 이다. 바꾸어 말하면 그것은 《정식으로》 출판할수 있는 준비가 되여 있지 않다. 그러나 이 책을 쓸 당시 지원기능은 http://www.ximian.com에 있는 Ximian 로부터 구입할수 있었다.

Gnumeric 에는 일반작업판에 대한 많은 도구들이 들어 있다. Gnumeric 에 대한 더 상세한 정보는 《pending tasks》의 일반적인 목록을 포괄하고 있는 http://www .gnome.org/gnumeric 에서 볼수 있다. 흥미 있으면 이 작업판을 개발하는데 참가할수 있다.

Gnumeric 의 리용

Gnumeric 의 리용에 대해 간단히 고찰하자. 이 작업판은 Red Hat 7.1 을 비롯한 수많은 표준 Linux 배포물의 한 부분이다. 다음과 같은 지령으로 xterm 창문에서 그것을 실행시킬수 있다.

```
$ gnumeric &
```

혹은 또 다른 지령을 리용할수 있다.

```
$ /usr/bin/gnumeric &
```

다른 방법으로 Main Menu 단추를 찰칵하고 Programs → Applications → Gnumeric 를 선택하여 Gnumeric 를 실행시킬수도 있다.

Gnumeric 는 그림 9-19 에서 보여 준것과 같은 빈 작업판으로 시작한다. 다른 표처리프로그람들과 마찬가지로 Gnumeric 에는 간단한 방법으로 공식을 편집하고 작성하는 각이한 도구에 봉사하는 수많은 아이콘들이 있다.

이제 File → Open 지령을 선택하면 Gnumeric 파일을 열수 있다. 이것으로 해서 필요한 파일 즉 Gnumeric XML 파일형태로 조종할수 있는 Load File 대화칸에 대해서 정통할수 있다.

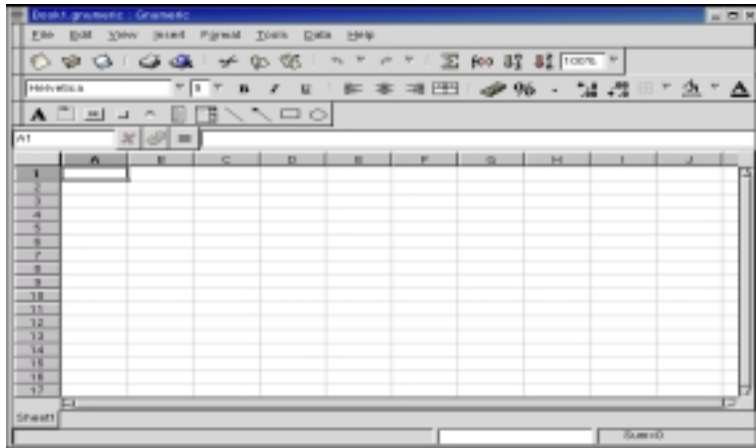



그림 9-19. Gnumeric 작업판

파일반입

파일차림표의 수많은 다른 파일형태로부터 Gnumeric 자료를 반입할수 있다. File → Import 를 선택하고 한개 파일을 선정하면 Gnumeric 는 다음과 같은 5 개 부류중의 하나로 그 파일을 배치하도록 입력재촉문을 내보낸다.

HTML File Made By Gnumeric Gnumeric 로부터 HTML 파일과 같은 작업판을 보관했으면 HTML 파일에서 자료를 다시 반입할수 있다.

Text File Import 자료를 배포하는 일반적인 한가지 방식은 본문파일인데 거기에서 반점들은 매개 자료항목을 구별한다. 이것은 한때 반점구획형태로 알려 졌다. Text File Import 보조항목은 Gnumeric 가 각이한 요소들에 자료를 할당하는데 매 반점을 리용하도록 한다.

SC/xspread File Import Gnumeric 는 Microsoft Excel, IBM(Lotus)1-2-3 그리고 Applixware 을 비롯하여 다른 형태의 작업판으로부터 자료를 반입할수 있다.

Data Interchange Format(DIF) Import Data Interchange Format 는 자료기지와 작업판사이에서 자료를 교환하기 위한 일반적인 수단이다. 그것은 또한 량립할수 있는 다른 파일형태들이 없을 때 작업판사이의 일반적인 변환수단으로 리용될수 있다.

Experimental Gnumeric 이 부류는 이미 Gnumeric 에 있고 .gnumeric 확장자를 가진 파일로부터《반입》할 때 리용된다.

주의	Gnumeric 의 그 후판본들에서도 역시 text 파일을 text 로 반입할수 있다. Text 파일의 매 행은 련속적인 cell 로 결합된다.

파일반출

다른 작업판을 리용하는 사람들과 함께 Gnumeric 파일을 공유할수 있다. Gnumeric 는 14 개의 서로 다른 파일형태에 반출을 지원한다. 이 처리를 진행하자면 File → Save

As 를 선택해야 한다. 선택항목들을 보기 위하여 그림 9-20 에 보여 준것과 같은 File Format 단추를 찰각해야 한다. 파일형태를 선정한 다음 OK 를 찰칵한다.

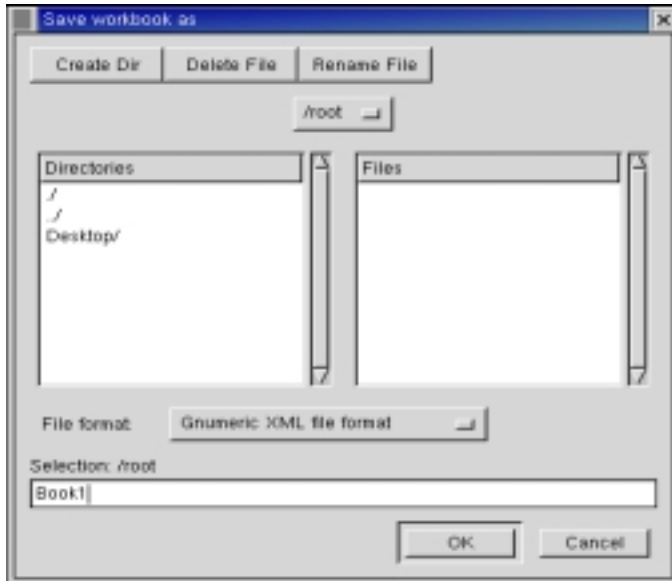



그림 9-20. Save Workbook As 창문

주의 각이한 프로그람들사이에서 파일을 교환하는 능력은 일반적으로 완전하지 못하다. 이것은 앞서 본 Excel 과 Gnumeric 를 보면 명백하다. 일부 사용자들이 2000 개의 Excel 파일을 Gnumeric 에 성공적으로 반입하였지만 이 두 프로그람의 모든 기능은 량립할수 없다.

규칙

표처리프로그람에서 심장부는 규칙이다. 정의에 의하면 규칙은 작업판에 세포단위로 정의되고 채워 진다. 규칙은 SUM 과 같은 기능을 가지는데 이 기능은 정의된 구역에 있는 요소들을 합계한다. Gnumeric 가 다른 작업판에서 파일을 바꿀 때 이 파일들의 기능들도 물론 바뀌여 진다. Gnumeric 기능을 잘 모르겠으면 Help → Gnumeric Function Reference 를 선택하여 방조를 받아야 한다.

X Windows 편의프로그람

이 장을 끝내기전에 몇개의 간단한 X Windows 편의프로그람과 그것을 리용하는 방법을 보자. 이 편의프로그람에는 다음과 같은것이 있다.

- gedit
- xclock
- xcalc

gedit 의 리용

gedit 는 Windows Notepad 와 비슷한 기능을 제공하는 GNOME 응용프로그람이다. 간단한 본문파일의 내용을 펼치고 변경시키기 위해서 본문을 탐색하고 gedit 를 리용하여 새로운 파일을 작성할수 있다.

의심할 여지없이 그림 9-21 에서 보여 준것처럼 gedit 대면부는 좀 간소하지만 간단한 본문파일들을 빨리 편집하는데 그것은 쓸데 없는 장식을 준다.

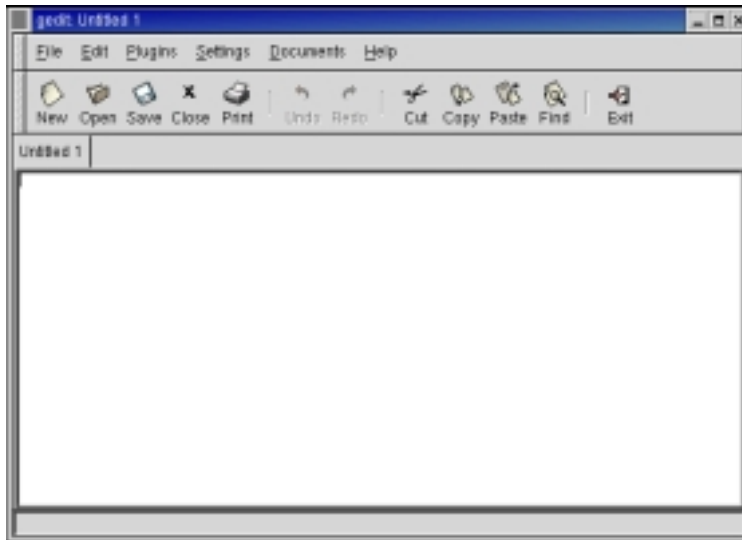



그림 9-21. gedit 창문

다음의 지령으로 gedit 창문을 기동할수 있다.

$ **gedit &**

혹은 gedit 가 경로에 없으면 다음의 지령을 리용할수 있다.

$ **/usr/bin/gedit &**

또는 GNOME 체계차림표를 찰칵해서 Programs → Applications → Gedit 를 선택할수 있다.

대면부는 간단하다. 우에 가로 놓인것은 파일을 열고 보관하는 단추와 본문을 복사하고 자르고 붙이고 탐색하는 단추가 있는 도구띠이다.

이미 존재하는 파일을 적재하기 위하여 Open 단추를 찰칵하거나 File → Open 을 선택한다. 여기서 여기저기 읽어 보고 펼치려는 파일을 선택할수 있다. 파일이 열리면 그것은 그림 9-22 에 보여 준것과 같이 나타난다.

도구띠의 Save 단추를 찰칵하거나 File → Save 를 선택하여 열었던 파일을 보관할수 있다.

도구띠의 Find 단추는 그림 9-23 에 보여 준 Find 대화칸을 제시한다. Edit → Find 를 선택하여도 이 대화칸을 역시 열수 있다.

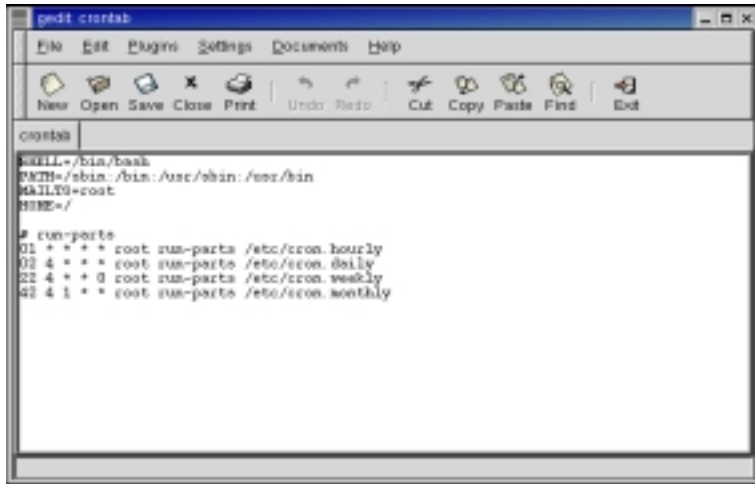



그림 9-22. 파일을 적재한 결과

이 대화칸의 리용은 상대적으로 단순하다. 즉 탐색하려는 본문을 써넣고 Find 단추를 찰칵한다. 또한 문서시작부터 탐색하겠는가 혹은 유표위치부터 하겠는가 그리고 어느쪽으로 탐색하겠는가를 지정할수 있다.

바꾸기는 한 단계 더 앞으로 나아가서 진행한다. Edit → Replace 를 선택하여 Replace 대화칸을 열수 있다. 이 대화칸은 찾은 매 항목을 바꾸는 본문을 지정하는 추가적인 마당을 가진다는것을 제외하고는 Find 대화칸과 비슷하다. 이외에도 본문을 바꾸기전에 입력해야 할것을 지정하기 위해서 Replace 단추를 리용할수 있다.

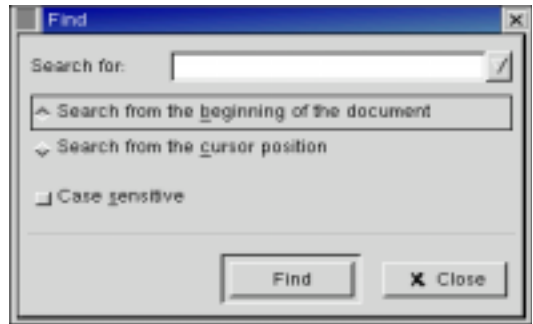



그림 9-23. Find 대화

X Windows 시계

GNOME 조종판에 나타난 시계를 보았을것이다. 일부 사용자들은 자기들의 탁상에서 큰 수자형시계나 상사형시계를 보려고 한다.

이것은 xclock 프로그람을 리용하여 보관할수 있다. xclock 는 상사형시계나 수자형시계를 만들수 있게 하는 기초적인 시계프로그람이다. 기정으로 xclock 는 상사형시계를 현시한다. 그러므로 다음의 지령을 리용하면 그림 9-24 와 같은 창문을 만들수 있다.

$ xclock &

혹은

$ /usr/x11R6/bin/xclock &

그림 9-24. 기정 xclock 형

Sun May 19 10:16:05 2002

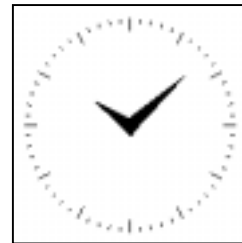
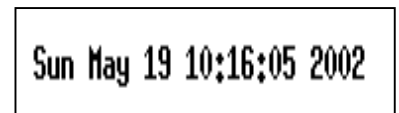



그림 9-25. 수자형 xclock 창문

-digital 기발을 리용하면 xclock 는 그림 9-25 에 보여 준것과 같은 수자형시계를 현시한다.

-digital 기발은 xclock 만이 인정하는 기발이 아니다. -hd 와 –hl 기발들은 상사형시계에 더 흥미 있는 시각현시장치를 만든다. -hd 기발은 시계의 바늘색을 정의하는데, –hl 기발은 시계바늘의 테두리색을 정의하는데 리용된다(이미전에 RGB 색에 대해서 론의한것을 기억하는가? 이것이 RGB 색을 어디에 리용해야 하는가에 대한 한가지 실례이다.).

그림 9-26. 시계바늘의 색갈선정

례를 들어 지령

$ xclock -hd yellow -hl green &

은 그림 9-26 과 같이 노란색바늘과 풀색테두리를 가진 시계를 만든다(물론 책에서 보기에는 회백색으로밖에는 보이지 않는다.).

그림 9-27. 배경색설정

시계의 바늘과 흰면사이 대조도가 약하다.이것은 표준적인 X Windows 기발 즉 xterm 에 대한 설명에서 이미 본 –bg 기발을 리용해서 수정할수 있다. 다음의 지령으로 백색을 검은색으로 정의하여 그림 9-27 과 같이 만들수 있다.

$ xclock -hd yellow -hl green –bg black &

그림 9-28. 전경색설정

바늘이 명백하게 보이게 되였지만 새로운 문제가 생겼다. 시계의 점을 가리키는 표식들이 보이지 않는다. 그것들은 검은색배경에 검은색으로 그려 졌다. 이 표식을 전경부분으로 만들고 색갈은 표준 X Windows 의 –fg 기발을 써서 바꿀수 있다. 다음의 지령을 리용하면 그림 9-28 과 같은 결과가 나타난다.

$ xclock –hd yellow –hl green –bg black –fg cyan &

GNOME 수판의 리용

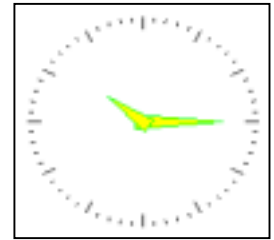
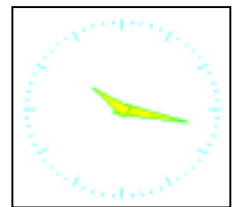
GNOME 을 포함한 Linux 의 모든 배포물에서 리용할수 있는 기정 GNOME 수판인 GNOMR Calculator 에 대해 간단히 고찰하자.

이 시점에서 기대할수 있는바와 같이 다음지령으로 GNOME Calculator 를 실행할수 있다.

$ gcalc &

혹은

$ /usr/bin/gcalc &

GNOME Calculator 는 기본 GNOME System 차림표의 Utilities 보조차림표에서

Simple Calculator 를 선택해서 열수도 있다.

GNOME Calculator 창문은 그림 9-29 와 같이 보인다.

Calculator 는 표준수산기와 같이 많은것들을 조작한다. 그것은 삼각함수와 로그함수를 비롯하여 필요한 수학함수들을 모두 제공한다. 이러한 수판과 그의 함수들에 대한 리용방법은 실제로는 이 장의 대상이 아니며 이 함수의 완전한 의미에 대한 설명은 수학책에서 보아야 한다.

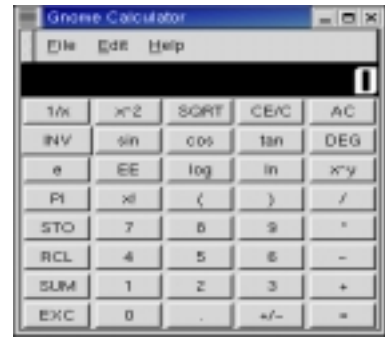



그림 9-29. GNOME Calculator 창문

▶미리 알아두기

이제는 GNOME 환경에 충분히 익숙되였다. GNOME 의 기초적인 선택항목들을 조종할수 있고 프로그람을 실행할수 있으며 창문을 다룰수 있게 되였다.

앞에서 서술한바와 같이 X Windows 와 GNOME 의 능력은 유연성과 구성으로 얻어 진다.

제 10 장 《고급한 GNOME 구성》에서는 개인적인 요구를 충족시키는 GNOME 의 구성에 대한 방법을 본다. 이 장에는 기본 GNOME 조종판의 전용화, 탁상출현과 주제들의 변경 그리고 기타 일반 GNOME 구성과제들의 수행 등이 포함된다.

제 10 장. 고급한 GNOME 구성

지금까지 GNOME 을 리용하는 방법과 GNOME 탁상환경에서 응용프로그람을 실행하는 방법을 배웠다. 이 장에서는 GNOME 에서 리용할수 있는 고급한 선택항목들을 고찰한다. GNOME 구성에서는 GNOME 조종쎈터(Control Center)가 기본이다. GNOME 조종쎈터에서는 다음과 같은 작업을 진행할수 있다.

- 도형방식대면부를 리용하여 탁상배경과 주제를 구성한다.
- 체계에 대한 다매체특성들을 정의한다.
- 대화칸 및 기타대면부 항목들을 마음대로 조정한다.
- Sawfish 창문관리기를 구성한다.

GNOME 조종판은 사용자대면부의 중심이다. GNOME 조종판은 기본 GNOME 조종판뿐아니라 사용자가 추가한 임의의 조종판의 위치, 크기, 동작, 기타 설정들을 조종할수 있는 개별적인 구성처리프로그람을 가진다.

이 장에서는 두 측면에서 GNOME 구성을 고찰한다.

GNOME 조종쎈터의 리용

GNOME 조종쎈터는 두가지 방법으로 호출할수 있다.

- 기본 GNOME 조종판우에 있는 Toolbox 단추를 찰칵한다.

- GNOME 의 기본체계차림표에서 Programs → Settings → GNOME Control Center 항목을 선택한다.

조종쎈터가 기동되면 처음에 그림 10-1 과 같은 창문이 나타난다.
이 창문은 두개 부분으로 되여 있다.

- 왼쪽은 7 개의 기본목록 즉 Desktop, Document Handlers, Multimedia, Peripherals, Sawfish Window manager, Session, User Interface 으로 이루어 진 구성항목들의 계층나무를 현시한다.
- 오른쪽은 선택된 계층나무의 개별적인 항목들에 대한 조종정보를 현시한다. 실례로 그림 10-2 에서는 Desktop Background 항목을 현시하였다.

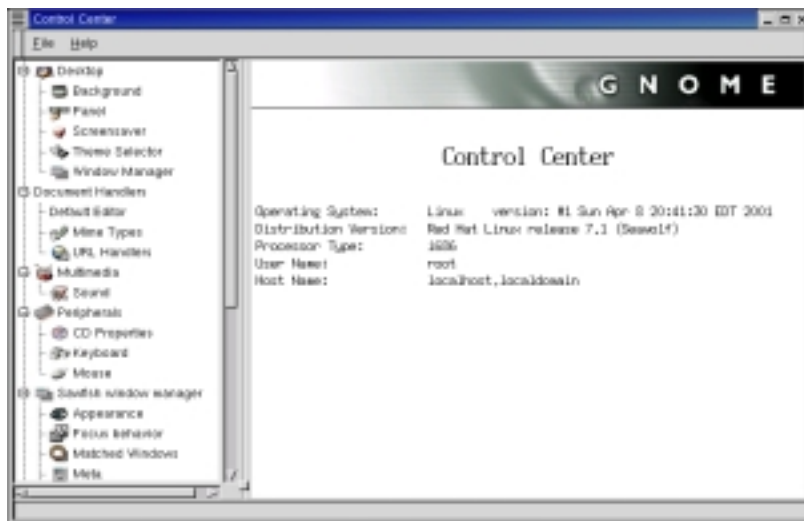



그림 10-1. 초기 GNOME 조종쎈터창문

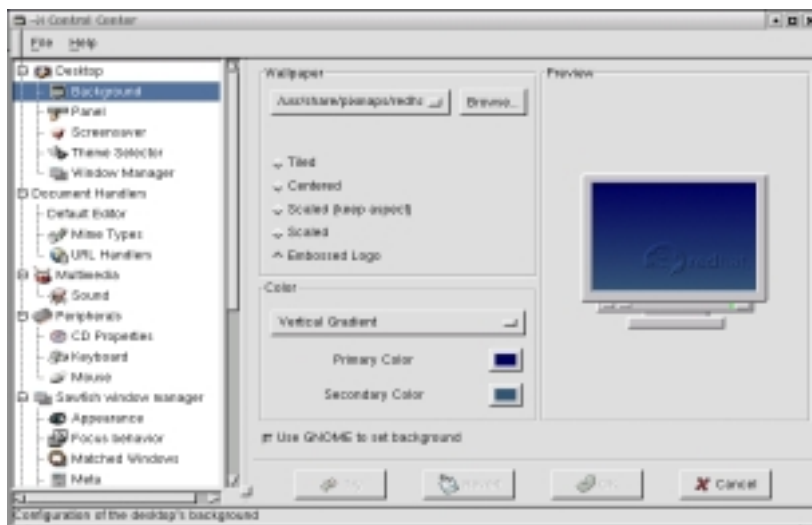



그림 10-2. 탁상배경부분

Desktop

조종쎈터의 Desktop 항목은 배경색과 벽지, 화면보호기(Screen Saver) 및 탁상주제의 선택을 비롯하여 탁상의 외관 및 동작과 관련한 여러가지 항목들을 조종할수 있게 한다.

배경속성들을 조종하기 위하여 조종쎈터창문의 왼쪽에 있는 Desktop 항목을 선택한다. 그러면 창문의 오른쪽에 그림 10-2에서 보여 준바와 같은 Desktop Background 애플레트가 현시된다.

이 창문에서 사용자는 3개의 탁상배경들을 만들수 있다.

탁상벽지 Browse 단추를 누르고 하드디스크로부터 배경벽지로 리용할 그림을 선택한다. 그 그림을 탁상중심에 놓거나 타일화할수 있으며 탁상전체에 전개할수 있다. 혹은 Embossed Logo 항목을 리용하여 배경에 Red Hat 문양을 나타낼수 있다.

단색 Color 내림목록에서 Solid를 선택하고 Primary Color를 선택하면 그림 10-3과 같은 Pick A Color 창문이 펼쳐 진다.

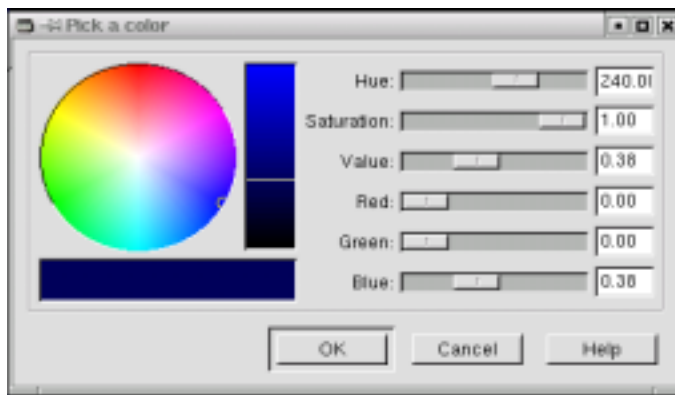



그림 10-3. Pick A Color 창문

두색그라디엔트 Color 내림목록에서 Vertical Gradient 혹은 Horizontal Gradient를 선택하고 기본색과 두번째 색을 선택한다. 그러면 기본색으로부터 두번째 색으로 변화되면서 화면의 우로부터 아래로(혹은 왼쪽으로부터 오른쪽으로) 그라디엔트의 효과가 나타난다.

배경설정변경이 완성되였으면 OK 단추를 찰칵해서 변경을 끝낸다.

조종판설정들을 조종하기 위하여 GNOME 조종쎈터의 탁상항목에서 Panel 항목을 선택한다. 그러면 GNOME 조종쎈터창문의 오른쪽에 그림 10-4와 같은 조종판구성을 위한 애플레트가 나타난다.

여기에는 조종판구성과 관련된 5개의 표쪽들이 있다. 이러한 설정으로 해서 조종판에 매 아이콘과 차림표의 보기, 감각, 동작 등을 구성할수 있다.

- Animation 표쪽은 GNOME 조종판으로부터 차림표들과 그림의 움직임을 조종한다.
- Buttons 표쪽은 조종판에서 매 프로그람기동단추의 감각을 조절한다. 매 단추의 깊이, 경계 및 상대적위치를 조종할수 있다. Tiles Enabled 항목을 선택하였을 때 Normal Tile 혹은 Clicked Tile을 선택하여 타일배경의 색과 패턴을 변화시킬수 있다.

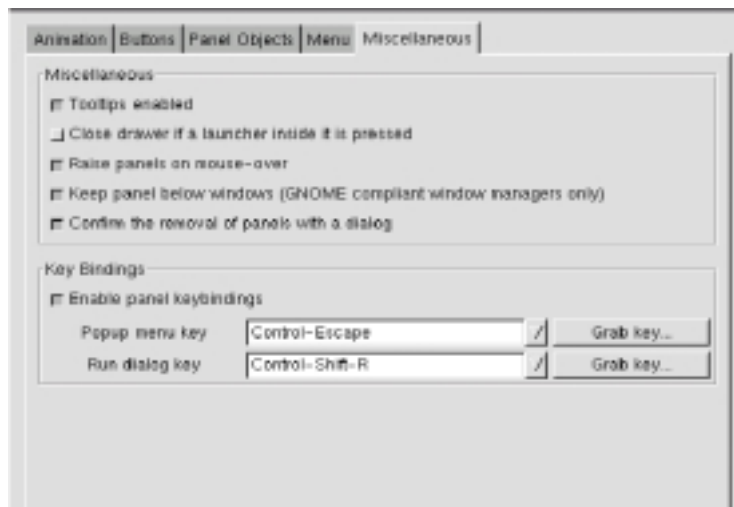



그림 10-4. 조종판구성

- Panel Objects 표쪽은 GNOME 조종판에 있는 객체들이 움직일 때 일어 나는 사건들을 정의한다.
- Menu 표쪽은 아이콘과 차림표의 크기 그리고 기본차림표로부터의 상대적인 위치를 조절한다.
- Miscellaneous 표쪽은 지정된 단추에서 마우스를 움직일 때 프로그람의 간단한 설명을 달아 주는 Tooltips Enabled 와 같은 설정항목을 포괄하고 있다.

배경설정변경이 완성되였으면 OK 단추를 찰칵해서 변경을 끝낸다.

화면보호기설정들을 조종하기 위하여 GNOME 조종쎈터의 Desktop 항목에서 Screen saver 항목을 선택한다. 그러면 GNOME 조종쎈터창문의 오른쪽에 그림 10-5 에서와 같은 화면보호기를 구성하는 애플레트가 현시된다.

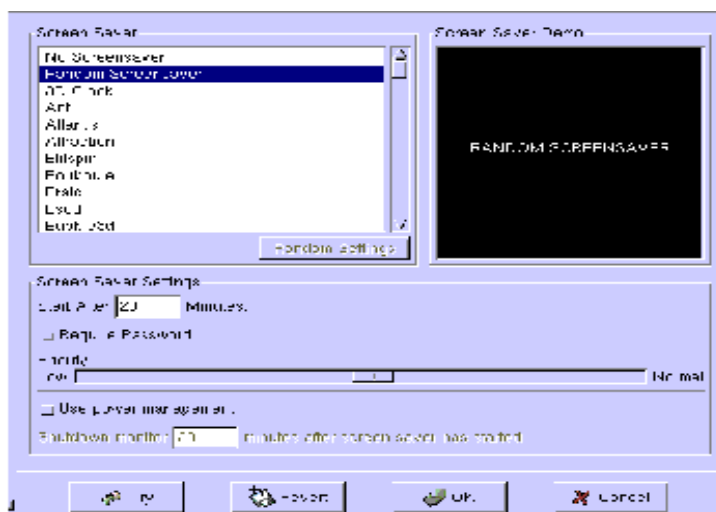



그림 10-5. 화면보호기구성

왼쪽 웃구석에 있는것은 자기의 체계에서 리용할수 있는 화면보호기들의 목록이다. 이 목록의 오른쪽에는 현재 선택된 화면보호기가 축소되여 4 각형구역안에 나타난다. 목록의 아래에는 몇분후에 화면보호기를 동작시키겠는가, 통과암호를 주겠는가를 지적한다. Priority 설정은 화면보호기에 콤퓨터자원을 어느 정도로 할당할수 있는가를 나타낸다. Priority 가 Normal 로 설정되면 배경처리들의 콤퓨터자원에 대한 호출은 금지되며 화면보호기프로그람이 원활하게 돌아 갈수 있다. 반대로 Priority 가 Low 로 설정되면 체계자원을 호출하는데서 배경처리들이 우선권을 가지므로 화면보호기도형들은 원활하지 못하다.

마지막으로 화면보호기가 동작중에 있을 때 전원관리기가 현시장치전원을 끌수 있게 하겠는가 그리고 화면보호기프로그람이 동작해서 몇분후에 현시장치의 전원을 끄겠는가 하는것을 설정할수 있다.

일단 설정이 끝나면 역시 OK 단추를 눌러 설정값들을 보관해야 한다.

조종쎈터의 Desktop 항목에서 또 다른 하나의 특성은 그림 10-6 에서 보여 준 Theme Selector 이다.

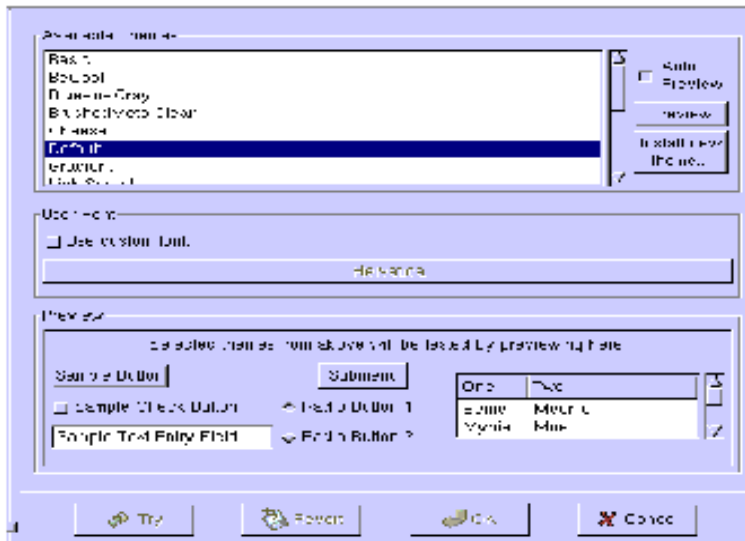



그림 10-6. 주제선택기

이 항목에서 사용자는 창문의 제일 웃부분에 있는 탁상주제의 목록가운데서 임의의 하나를 선택할수 있다. 이러한 탁상주제들은 창문배경, 단추양식, 창문양식과 색구성들에 대한 종합적인 설정들을 정의한다. Use Custom Font 를 선택하여 주제를 조정할수 있다. 서체(그림 10-6 에서는 Helvetica 임)를 찰칵하면 Pick A Font 창문이 펼쳐 지는데 이 창문에서 서체, 양식, 크기들을 선택할수 있다. 주제가 선택되면 그림 10-7 과 같이 애플레트절반아래에 주제의 실례가 표시된다. 여기서 Quiet 주제가 선택되였다.

조종쎈터의 Desktop 항목의 마지막특성은 그림 10-8 에 보여 준 Window Manager 선택기이다.

이 항목에서는 가능한 모든 창문관리기들을 보여 준다. 이 차림표에서 창문관리기를 절환할수 있다. Run Configuration Tool For Sawfish 선택항목은 GNOME 조종쎈터

의 Sawfish 창문관리기항목에서 보여 준것과 같은 선택항목들을 구성할수 있는 또 다른 창문을 펼친다. 이러한 선택항목들은 이 장의 마지막에서 고찰하게 된다.

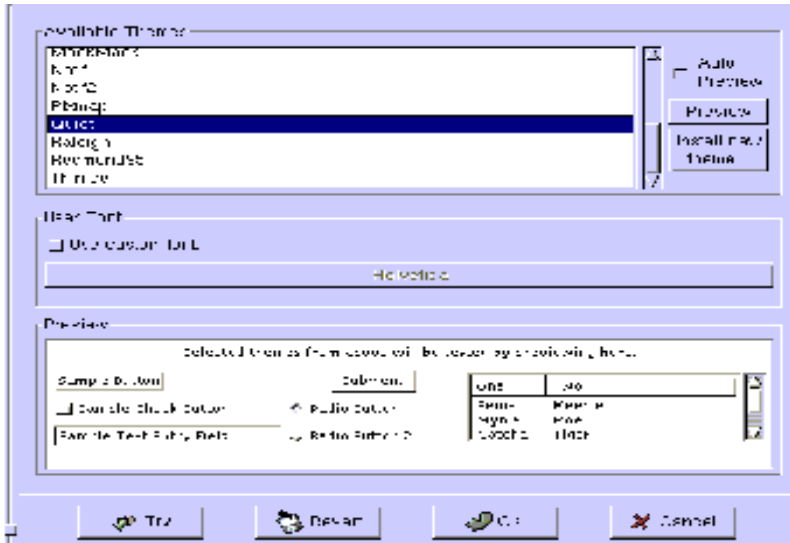



그림 10-7. Quiet 주제선택

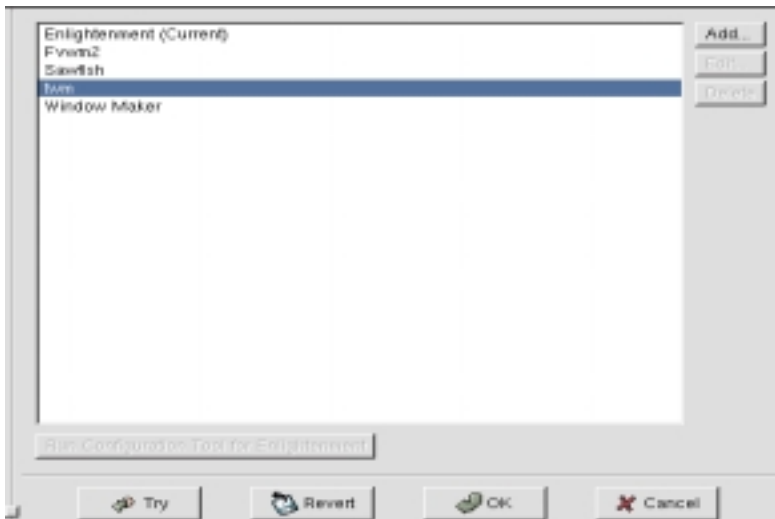



그림 10-8. Window Manager 선택기

Document Handlers

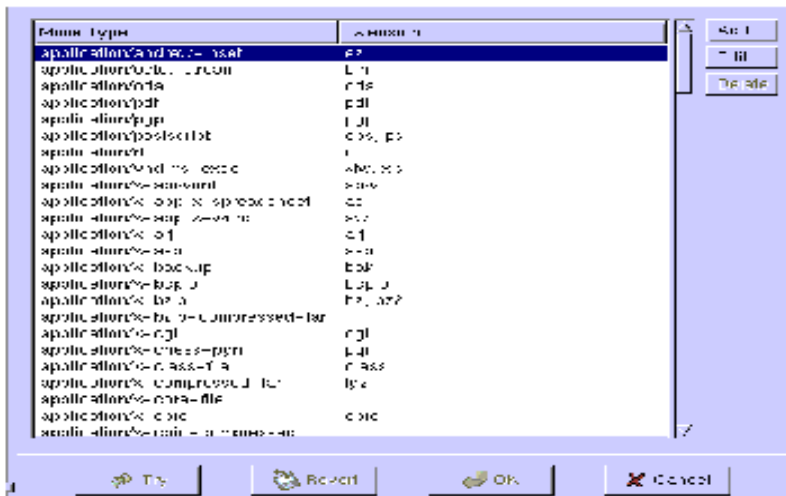
Document Handlers 항목은 여러 형태의 파일과 관련된 응용프로그람을 관리하는데 이것은 기정본문편집기, 기타 응용프로그람을 위한 가상형태(Mime types)와 Web 주소를 위한 URL 을 포괄하고 있다. 다시말하여 여기서는 GNOME 파일관리기의 자료파일을 열 때의 응용프로그람이 설정된다.

본문파일들과 관련된 편집기를 설정하기 위하여 Default Editor 항목을 선택한다. 그러면 그림 10-9 와 같이 구성을 위한 애플레트가 현시된다.

Gnome Editor 내림목록칸을 찰칵해서 기정편집기를 선택한다. Emacs 는 기정 Gnome 편집기이다. 기정만이 있고 emacs 가 콤퓨터에 설치되지 않았을 때 GNOME 파일관리기를 리용하여 본문파일을 열면 오유통보가 난다.

그림 10-9. 기정편집기설정

문서확장자와 관련된 응용프로그람의 형태를 정의하기 위하여 Mime types 항목을 선택한다. Mime types 항목은 그림 10-10 과 같이 구성을 위한 애플레트를 현시한다.

그림 10-10. 여러가지 가상형태

Linux 파일이 확장자가 없어도 각이한 형태의 파일로 나눌수 있다. 선택한 가상형태를 찰칵한 다음 Edit 를 찰칵한다. 펼쳐 진 Set Actions For 창문에서 사용자들은 가상형태와 관련된 아이콘, 확장자와 응용프로그람을 설정할수 있다.

Universal Resource Locators(URLS)를 리용하는 Web 페지나 파일과 관련된 응용프로그람의 형태를 정의하려면 URL Handler 항목을 선택해야 한다. 이 항목은 그림 10-11 에서 보여 준 구성을 위한 애플레트를 현시한다.

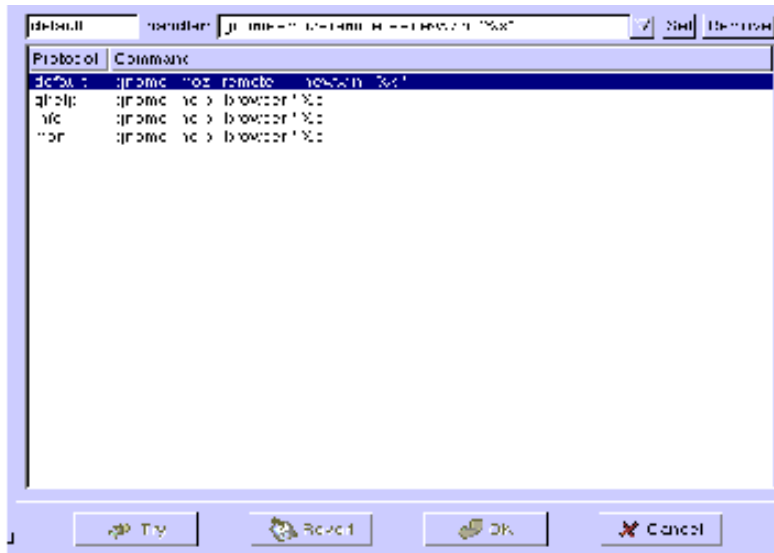



그림 10-11. URL 조종기

그림 10-11 에서와 같이 4 개 류형의 파일이나 규약이 있는데 그것은 URL 로 설정된다. URL 이 대부분 Web 페지와 련관되여 있지만 GNOME 역시 방조파일, 정보페지와 소개페지를 URL 로 지정할수 있게 한다.

기정으로는 두개의 열람기가 있는데 그것은 Netscape 와 GNOME 방조열람기이다. 이미 있는 열람기나 새로운 창문중 어느 하나로 열수도 있다. 변경하려는 규약을 선택해야 한다. 선택들을 조사하려면 Handler 창구의 내림화살단추를 누르면 된다. 선택한 다음에 Set 를 찰칵한다.

Multimedia

Multimedia 부분에서는 GNOME 에 기초한 체계에서 두가지 음성형태를 조종할수 있는데 GNOME 을 시작할 때 음성체계가 동작하는것과 음성사건(Sound event)을 만드는 동작으로 갈라 진다.

음성기동순서를 정의하려면 왼쪽조종판에서 Multimedia 에 있는 Sound 를 선택해야 한다. 기정으로 그림 10-12 에서와 같은 General 표쪽이 펼쳐 진다.

GNOME 창문에서 소리봉사를 제공 받자면 Enable Sound Server Startup 항목을 선택하여야 한다. 이 표쪽에서 음성사건을 발생시키기 위해 Sounds For Events 항목을 또 선택하여야 한다. 사건들이 발생할 때 울리는 소리를 정의하려면 그림 10-13 에서 보여 준 Sound Events 표쪽을 찰칵하여야 한다.

넓은 범위의 음성사건들을 재생하는 개별적인 음성파일(WAV 형식이며 이것은 Windows 체계에서 공통이다.)을 정의하기 위해서 Sound Event 표쪽을 리용할수 있다. 보여 준 매 사건들은 지정된 WAV 파일과 련관되여 있다. 하나의 사건을 선택하면 Play 단추를 찰칵해서 새로운 음성을 검사할수 있다. 대신에 Browsing 단추를 찰칵하고 리용하려는 여러가지 WAV 파일들을 조사해서 새로운 음성을 선택할수 있다.

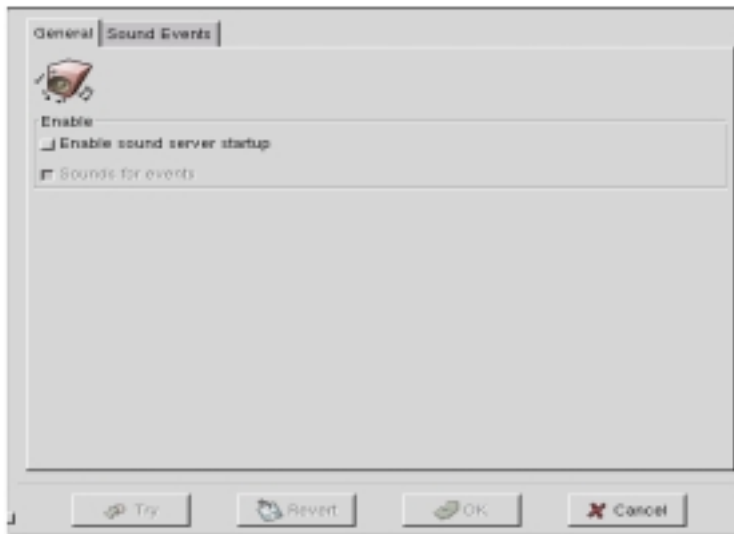



그림 10-12. 소리기동순서의 구성

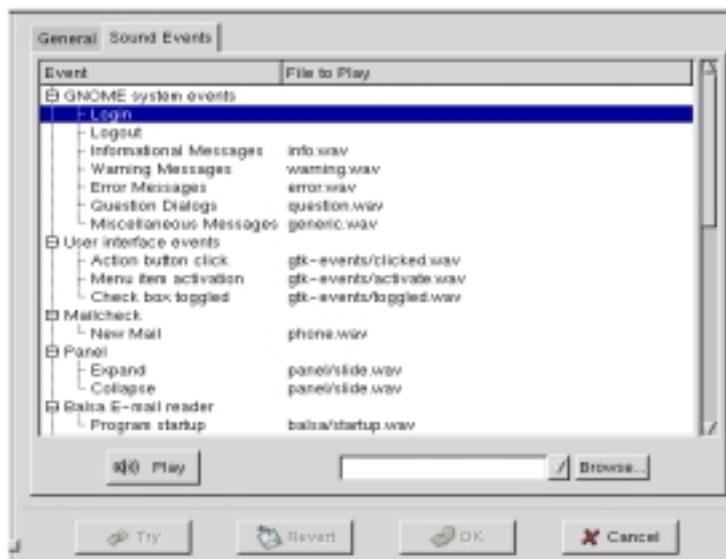



그림 10-13. Sound Event 표쪽

Peripherals

Peripherals 부분에서 CD Properties, Keyboard, Mouse 항목을 리용하여 CD 와 건반, 마우스의 동작형태를 조종할수 있다.

그림 10-14 에서 보여 준 CD Properties 항목은 CD 를 알맞게 구성된 CD 구동기에 넣었을 때 Linux 동작을 구성한다.

자료가 들어 있는 CD 를 넣은 경우에 세 선택항목가운데서 하나 혹은 모두를 구성할수 있다.

즉 기정구동기(Red Hat Linux 7.1의 /mnt/cdrom)에 CD 파일을 자동적으로 설치할수 있고 그 구동기의 기본등록부에 있는 프로그람을 자동적으로 실행할수 있으며 그 등록부에 대한 GNOME 파일관리기를 열수 있다.

음악 CD 를 넣은 경우에 Linux 가 CD 를 인식하면 전용지령을 실행하기 위해서 Linux 를 설치할수 있다. 기정으로 GNOME 조종쎈터는 GNOME CD Player 를 리용하여 어떤 음악이든지 재생할수 있게 설정된다. GNOME CD Player 지령선택항목에 대한 구체적인 정보를 보려면 xterm 을 열고 gtch-help 지령을 실행해야 한다.

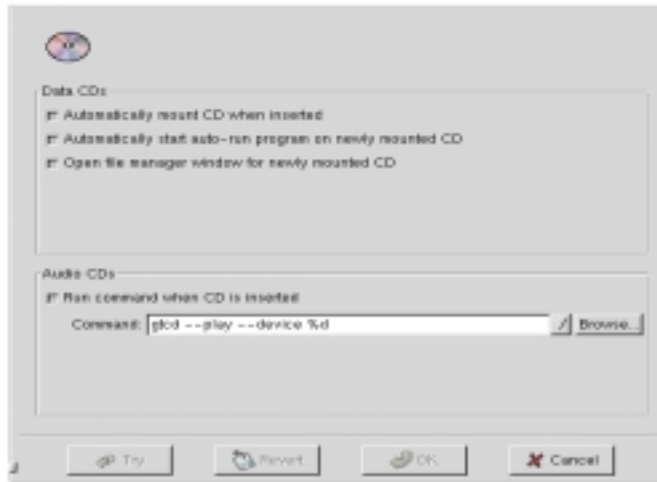



그림 10-14. CD 구동기의 구성

그림 10-15 에서 보여 준 Keyboard 항목은 건반자동반복기능을 조종한다. 여기서는 반복을 시작하기전에 시간을 규정하고 문자를 얼마나 빨리 반복시킬수 있는가를 교정할수 있다. 또한 건이 눌리울 때마다 내는 소리를 조종할수 있다. 건이 눌리웠을 때 나는 음량을 조절할수 있다. 또한 건반통보문이나 오유가 존재할 때 울리는 건반종의 음량, 주파수, 울림시간을 규정할수 있다.

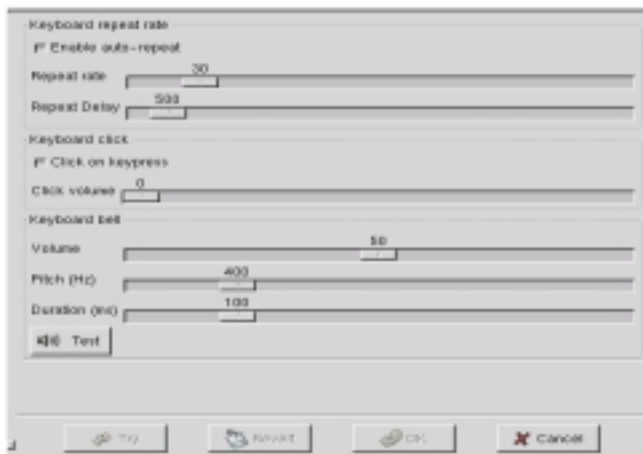



그림 10-15. 건반구성

그림 10-16 에서 보여 준 Mouse 마우스항목은 사용자가 왼손잡이인가 오른손잡이인가에 따라서 마우스의 왼쪽과 오른쪽단추의 기정동작을 효률적으로 바꿀수 있게 하며 마우스속도와 반응을 규정한다.

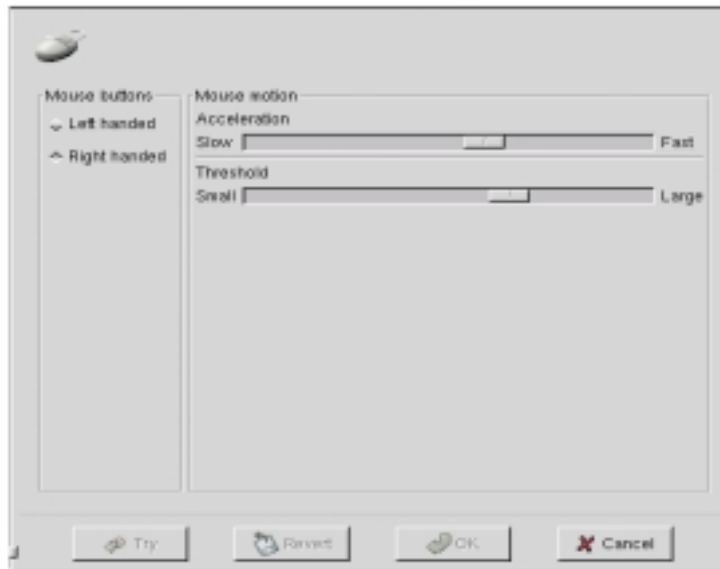



그림 10-16. 마우스의 동작설명

Sawfish 창문관리기

Red Hat Linux 7.1 에서 Sawfish 창문관리기가 기정으로 설치되였다고 보고 이 항목은 GNOME 탁상에서 도형사용자대면부에 《보고느낌》(look and feel)을 제공하는 구성을 조종한다. 각이한 선택항목을 열람한것과 같이 4 개의 단추들 즉 Try, Rervert, OK, Cancel 을 리용할수 있다. 변경한 다음에는 Try 를 눌러야 효과를 볼수 있다. Revert 를 누르면 본래의 구성으로 돌아 간다.

Sawfish 를 구성할수 있는 11 개의 항목이 있다.

- Appearance 는 기정틀형태를 설정한다. Red Hat Linux 7.1 에서 GNOME 기정틀형태는 CoolClean 으로 알려 져 있다. 서체, 대화칸, 동화(animations)들은 사용자가 요구하는대로 틀형태를 전용화할수 있다.
- Focus Behavior 는 초점이 옮겨 간 창문의 동작상태를 규정한다.
- Matched Windows 는 창문에 여러가지 형태의 동작을 련결하게 한다.
- Meta 는 3 개의 항목 즉 초학자, 중간급전문가와 전문가를 포괄하는데 이것은 GNOME 조종쎈터에서 Sawfish 를 구성할수 있는 준위를 규정한다.
- Minimizing/Maximizing 은 최소화되였을 때 가상창문의 위치와 최대화되였을 때 창문동작을 조종한다.
- Miscellaneous 설정은 창문이 덧놓였을 때 다른 창문들의 특성과 매 창문과 관련된 쓸모 있는 도구묶음을 결정한다.
- Moving/Resizing 은 이동하거나 크기가 변할 때 창문의 움직임과 위치를 결정한다.

- Placement 설정은 지정된 프로그람에 관한 대화칸이 있는 창문이 열릴 때 위치를 결정한다.
- Shortcuts 는 GNOME 에서 탈퇴와 같은 특수한 지령과 관련되는 건결합을 포함한다.
- Sound 설정들은 최소화, 최대화와 같은 창문사건들에 소리효과를 준다.
- Workspaces 는 가상탁상들의 수와 각이한 가상탁상환경에로 넘어 가는데 필요한 동작을 결정한다.

보통 Sawfish 를 구성할수 있으므로 이 부분의 선택항목은 콤퓨터에서 보게 되는 선택항목과 달라 진다. 다음지령들을 리용하여 Sawfish 사용자대면부에서와 같은 구성선택항목을 직접 얻을수 있는데 이것은 다음과 같은 지령행으로부터 기동시킬수 있다.

$ sawfish-ui &

혹은 어떤 원인으로 Sawfish 가 경로에 없으면 다음지령을 리용해야 한다.

$ /usr/bin/sawfish-ui &

이 지령을 실행하면 그림 10-17 에서 보여 준 Sawfish 창문관리기사용자대면부가 나타난다. 초학자, 중간급, 전문가급사용자들을 위한 Meta 선택항목들이 처음에 Sawfish 구성창문에 나타난다는것을 명심해야 한다.

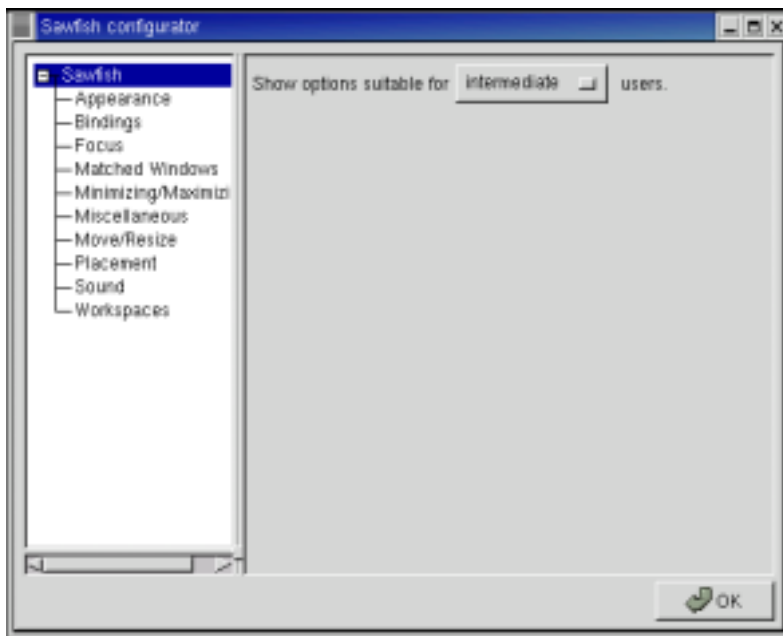



그림 10-17. sawfish 창문관리기구성

Session

Session 항목은 GNOME 이 기동할 때 발생하는 사건들을 조종한다. Startup Hint 선택항목은 가입할 때 보게 되는 암시나 통보들을 구성한다. Startup Programs 선택항목은 탁상관리기에 가입할 때 실행하는 프로그람과 그것들의 동작방식을 구성한다.

User Interface Options

User Interface Options 항목은 다음과 같은것들을 포함하는 여러가지 형태의 사용자대면부를 조종한다.

- GNOME 응용프로그람에 대한 차림표, 도구띠와 같은 창문요소들의 기정 외관
- 대화칸요소들의 표시방법
- 같은 응용프로그람으로 다중창문을 조종하는 방법

이 장에서 구체적으로 론의하지 않은 조종쎈터의 구성설정 및 기타 설정들의 리용법에 대해서는 GNOME 방조체계를 리용해야 한다. GNOME 체계의 기본차림표로부터 방조체계를 선택하여 GNOME 방조체계를 호출할수 있다.

GNOME 조작판의 구성

GNOME 사용자대면부에서 기본은 GNOME 조종판이다. GNOME 을 처음으로 기동하면 기정조종판이 만들어 진다.

사용자들은 자기의 조종판을 만들수 있다. GNOME 기본차림표로부터 Panel → Create Panel 을 선택해서 5 가지 조종판가운데서 하나를 열수 있다. 즉.

- Menu 조종판은 GUI 창문에 내림차림표를 포함하는 오직 한개의 Menu 조종판을 추가할수 있다.
- Edge 조종판은 GUI 창문의 각이한 변두리에 조종판을 추가한다.
- Aligned 조종판은 단추에 필요한것만큼 더 추가하는것을 제외하고는 Edge 조종판과 같다.
- Sliding 조종판은 창문구석 혹은 중심에 배치하지 말아야 한다는것을 내놓고는 Aligned 조종판과 같다.
- Floting 조종판은 화면우의 임의의 곳에 배치할수 있는 조종판이다.

일단 조종판의 속성을 작성하면 변경시킬수 있다. 조종판에서 마우스의 오른쪽단추를 찰칵하면 Popup 차림표가 나타나는데 거기서 Panel → Properties → All Properties 를 선택한다. 현시된 속성창문에서 Menu 조종판을 제외하고는 임의의 조종판의 위치와 색갈 등을 조절할수 있다.

또한 다른 방법으로 GNOME 조종판을 구성할수 있다. 이 절에서는 기정 GNOME 조종판으로 작업할수 있다.

조종판에서 수행하려는 기본구성작업은 다음과 같다.

- 서랍(drawer)을 추가
- 탈퇴(logout)단추를 추가
- 시동기(launcher)를 추가
- 조종판속성을 조종

조종판에 서랍을 추가

조종판배경에서 오른쪽찰칵하여 나타나는 차림표로부터 Panel → Add To Panel → Drawer 를 선택하여 조종판에 한개 서랍을 추가한다.

서랍을 열거나 닫으려면 그것을 찰칵해야 한다. 열린서랍은 응용프로그람을 실행시키는것과 같은 다른 아이콘을 표시하고 포함할수 있다. 서랍이 닫기면 아이콘들은 숨겨 진다. 열린 서랍에 옮기려는 항목을 찰칵하여 조종판에서 아이콘과 애플레트를 옮길수 있다. 그다음 나타난 차림표에서 Move 를 선택하고 유표를 움직여 항목을 필요한 위치에 끌어 간다. 마지막으로 옮기기를 끝내기 위하여 그 위치에서 찰칵한다.

조종판에 탈퇴단추추가

보통 GNOME 기본차림표에서 Logout 를 선택하여 GNOME 으로부터 탈퇴할수 있다. Logout 를 한번 찰칵하여 실행되도록 조종판에 단추를 추가한다. 이를 위하여 조종판배경에서 찰칵한 다음 Panel → Add To Panel → Logout 단추를 찰칵한다. 그러면 Logout 단추가 조종판에 추가된다.

새로운 시동기의 추가

기정 GNOME 조종판에는 Netscape, 말단창문과 같은 응용프로그람을 기동하는 여러개의 단추들이 있다. 바로 이것들이 시동기단추들이다. 조종판배경에서 찰칵하여 나타난 차림표에서 Panel → Add To Panel → Launcher 를 선택하여 기동단추를 만들수 있다. 그림 10-18 과 같은 대화칸을 펼치고 기동하는 응용프로그람을 지정한다.

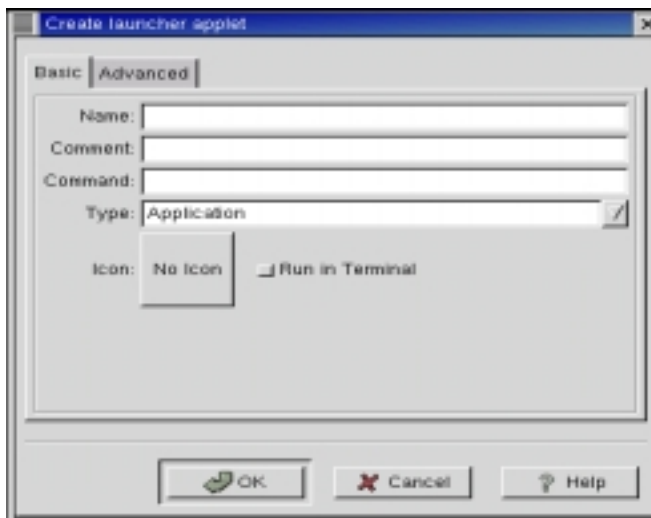



그림 10-18. 새 응용프로그람시동기의 지정

적어도 시동기의 이름, GNOME 혹은 X Windows 응용프로그람의 경로와 이름을 주어야 한다. 선택하려는 아이콘묶음을 표시하기 위하여 No Icon 이라고 표시된 단추를

찰각하여 시동기의 아이콘을 선택할수도 있다.

조종탁응용프로그람(지령행 혹은 말단창문에서 실행되는 프로그람)을 실행하는 시동기가 필요하면 Run In Teriminal Check 칸을 선택해야 한다.

조종판속성을 조종

조종판배경에서 찰각하여 Panel → Properties → All Properties 를 선택하면 그림 10-19 와 같은 조종판속성대화칸들이 현시된다.

이 대화칸에는 두개의 표쪽이 있다. Edge 조종판표쪽은 화면에서 조종판의 위치, 조종판의 크기(그리고 련관된 아이콘), 자동숨김조종과 조종판의 량쪽 끝에 있는 숨김단추들의 상태를 조종한다. 이 특성들은 제 8 장 《GNOME 과 X Windows 의 리용》에서 이미 고찰하였다.

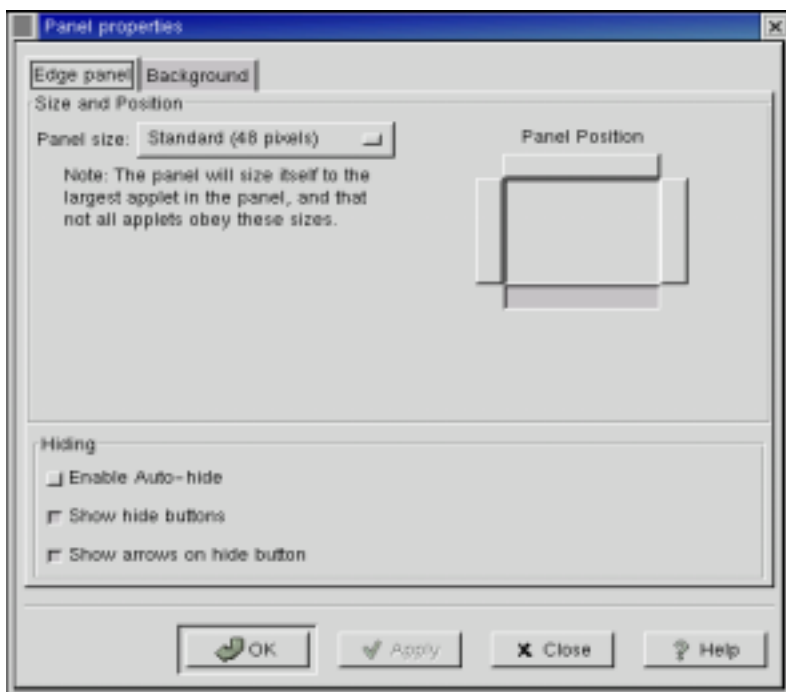



그림 10-19. 조종판속성대화칸, Edge 조종판표쪽

Background 표쪽을 그림 10-20 에서 보여 주었다. 여기서는 조종판의 배경그림이나 색갈을 선택할수 있다. 화면 쪽대기에 있는 Standard, Pixmap, Color background 중에서 선택할수 있다. Standard 배경은 GNOME 을 위해 정의한 주제 혹은 색도식에 따르며 Pixmap 는 배경으로 bitmap 영상을 리용하고 Color 는 배경의 고정적인 색을 지정한다.

Pixmap 가 선택된 경우 Browse 단추를 찰각해서 나타난 파일선택대화칸으로부터 그림파일을 선택할수 있다. 기정으로 그림을 잡아 늘쿨수도 있고 회전(Edge 조종판이 수직인 경우)시킬수도 있다.

Color 를 선정하면 Color To Use 단추를 찰각하여 요구하는 조종판배경색을 선택할 수 있는 Pick A Color 창문을 열수 있다.

그림 10-20. 조종판속성대화칸, Background 표쪽

▶미리 알아두기

지금까지 GNOME 과 X Windows 에 대하여 보았다. 제 11 장 《KDE 의 리용》에서는 KDE 라고 하는 도형사용자대면부를 보게 된다. KDE 의 력사를 고찰하고 그다음 그것을 설치하는데로 넘어 간다.

일단 KDE 탁상환경이 있고 기동하면 그것의 특성을 알수 있고 파일체계를 열람할수도 있다. 기초적인 문제들을 리해한후에 필요에 따라 KDE 를 구성하는 방법을 배우게 된다. 이 장의 마지막에 KDE 를 리용한 본문과 그림의 보기와 시간관리, Linux 체계의 관리방법을 배우게 된다.

제 11 장. KDE 의 리용

Linux 사용자대면부에서 가장 유망한 개발제품의 하나가 바로 K 탁상환경(KDE 즉 K Desktop Environment)이다. KDE 는 령으로부터 개발된 완전한 도형방식탁상환경이다. KDE 는 또한 가장 주요한 Linux 배포물의 기정탁상환경이기도 하다. 이 책에서 설명하는 Red Hat Linux 7.1 은 기정으로 GNOME 탁상을 설치하기는 하지만 KDE 판본 2.1.1 에 필요한 파일들도 포괄하고 있다.

KDE 대상과제의 력사

Linux 배포물의 다른 구성요소들과 마찬가지로 KDE 도 역시 무엇인가 새롭고 쓸모있는것을 만들어 내려고 하는 헌신적인 프로그람개발자들이 개발한것이다. 도이칠란드의 마티아스 어트리취(Matthias Ettrich)는 1996 년 10 월에 KDE 대상과제를 착상하였다.

기본목적은 Solaris 와 같은 업무적인 Unix 가동환경에 리용되는 업무적성격을 띤 일반탁상환경(CDE)과 류사한 강력한 탁상환경을 Linux 와 기타 Unix 체계에서 리용할수 있게 만드는것이였다. 결과 CDE 의 단순한 모방이 아니라 훨씬 더 넓은 범위에서 리용할수 있는 대상과제로 되였다.

KDE 대상과제에는 전자우편을 리용하여 세계 수백명의 개발자들이 참가하고 림시적인 자칭협회도 있다. KDE 개발에 참가하려는 개별적사람들은 창문관리기, 탁상조종판, 기타 구성요소와 같은 핵심적인 구성요소로 작업하거나 KDE 가 정의한 응용프로그람대면부(API)를 리용하여 KDE 에 필요한 자기식의 응용프로그람을 만들수 있다.

KDE 대상과제에 공헌한 사람들속에는 체계 및 응용프로그람작성가들외에 번역원, 문필가, 사용자대면부설계가 및 다매체전문가들도 있다.

Linux 콤퓨터에서 KDE 는 X Windows 를 대신하지 못한다. 대신에 KDE 는 탁상을 만들기 위한 도형처리기술로서 X Windows 체계를 리용할수 있다.

Sawfish, Afterstep 지어 CDE 와 같은 다른 X Windows 관리기를 써보면 KDE 가 kwin 이라고 하는 자기자체의 창문관리기를 가지고 있다는것을 알수 있다. 일부 창문관리기기능에는 탁상배경에 아이콘을 배치할수 있는 기능도 있다.

KDE 는 화면에 창문들을 만들고 관리하는 핵심기능들을 제공하는 kwin 창문관리기외에 응용프로그람들을 쉽게 호출할수 있게 하는 Taskbar 와 기본차림표와 같은 몇개의 주요구성요소들을 포함한다.

마지막으로 KDE 배포물은 표준편의프로그람, 체계관리도구, 오락프로그람제품과 같은 수많은 응용프로그람묶음을 가지고 있다. 지어 GNOME Office 와 Microsoft Office 2000 과 류사한 통합사무용프로그람들도 있다.

KDE 배포물의 고찰

Linux 와 마찬가지로 KDE 에서는 탁상구성요소들의 완전한 모임과 그것들을 동반하는 기정응용프로그람묶음을 서술하기 위하여 배포물이라는 말을 리용한다.

KDE 배포물은 다음의 구성요소들을 포함한다.

- 모든 KDE 응용프로그람들이 리용하는 도형처리서고
- 모든 KDE 응용프로그람들이 리용하는 핵심적인 KDE 서고
- 대부분의 응용프로그람들이 리용하는 제공함수묶음
- 창문관리기와 조종판/과제띠프로그람들과 같이 KDE 를 정의하는 기초응용프로그람
- 실행하고 있는 KDE 체계에 설치할수 있는 선택응용프로그람

Linux 와 관련된 대부분의 대상과제들과 마찬가지로 KDE 에 대한 원천코드들도 내리적재하여 고찰할수 있다. KDE 코드는 GNU 일반공개사용허가(GPL)밑에서 리용할수 있다.

KDE 응용프로그람의 작성

KDE 는 두해사이에 령으로부터 특색 있는 탁상으로 발전하였다. 이것은 노르웨이의 오슬로에 있는 Troll Tech 가 이미 개발한 업무용도형처리도구함인 Qt 를 리용하여야만 쓸수 있었다.

주의 Qt 를 리용하여 KDE 개발자들은 대면부를 표준화하기 위한 도구함을 만드는데 시간을 소비하지 않고 탁상을 위한 설계문제들에 집중해서 코드작업을 즉시 시작할수 있다.

Qt 는 KDE API 의 기초를 이루는 교차가동환경도형처리도구함이다. KDE 용으로 작성한 응용프로그람들은 Qt 기능을 리용할수 있다. 결과 완전한 KDE 응용프로그람이 며칠내에 완성될수 있다.

주의 Qt 가 GNU 일반공개사용허가밑에서 Troll Tech 에 의해 최근에 다시 발표됨으로써 Linux 세계에서 KDE 를 리용하는 사용자수가 계속 증가하고 있다.

Qt 와 KDE 는 둘다 객체지향프로그람언어인 C++로 작성되였다. 높은 수준객체지향언어인 Python 이나 Perl 로도 작성할수 있다.

Kdevelop 와 KDbg 를 비롯하여 KDE 를 리용할수 있는 좋은 개발도구들이 몇개 있다. 더 상세한 정보를 보기 위해서는 KDE Developer Web 싸이트(http://develop.kde.org)를 조사하여야 한다.

KDE 와 Red Hat Linux 7.1

Red Hat Linux 7.1 에서 KDE 판본 2.1.1 을 실행하는데 필요한 파일들을 배포물로 리용할수 있다. Red Hat 설치과정에 KDE 를 설치했다면 체계에 KDE 가 이미 설치되였다고 볼수 있다. 그렇지 않으면 다음절에서 리용하게 될 RPM 파일을 /mnt/cdrom/Red Hat/RPMS 에 있는 하나의 Red Hat Linux 7.1 설치 CD-ROM 에서 찾을수 있다(/mnt/cdrom 에 CD-ROM 을 설치했다고 하면).

Red Hat Linux 7.1 CD-ROM 우에서 KDE 파일들은 다음과 같다.

- kdeadmin-2.1.1-3.i386.rpm
- kdebase-2.1.1.-8.i386.rpm
- kdebindings-2.1.1.-1.i386.rpm *
- kdebindings-devel-2.1.1.-1.i386.rpm *
- kdebindings-kmozilla-2.1.1.-1.i386.rpm *
- kdegames-1.1.2.-6.i386.rpm *
- kdegraphics-1.1.2.-6.i386.rpm *
- kdelibs-2.1.1.-5.i386.rpm
- kdelibs-devel-2.1.1.-5.i386.rpm *
- kdelibs-sound-2.1.1.-5.i386.rpm
- kdelibs-sound-devel-2.1.1.-5.i386.rpm *
- kdemultimedia-2.1.1.-1.i386.rpm *
- kdenetwork-2.1.1.-1.i386.rpm
- kdenetwork-ppp-2.1.1.-1.i386.rpm
- kdepim-2.1.1.-1.i386.rpm
- kdesdk-2.1.1.-1.i386.rpm *
- kdesdk-devel-2.1.1.-1.i386.rpm *

- kdesupport-devel-2.1.-3.i386.rpm *
- kdetoys-2.1.1.-2.i386.rpm *
- kdeutils-2.1.1.-1.i386.rpm
- kdoc-2.1.1.-1.noarch.rpm *
- koffice-2.0.1.-2.i386.rpm
- kppload-1.04-23.i386.rpm

별표(*)로 표시된 프로그람묶음은 이 책에서 언급하지 않은 두번째 Red Hat 7.1 설치 CD-ROM 에 있다. http://www.kde.org 에 있는 KDE 홈페지나 http://www.rpmfind.net에 있는 RPM 자료기지싸이트로부터 그것들을 내리적재할수 있다.

KDE 의 갱신

이 책에서 언급한것처럼 Red Hat Linux 7.1 은 최신판본의 KDE 를 포함한다. 그러나 KDE 는 몇달에 한번씩 탁상에 있는 특성들을 갱신하므로 사용자들은 자기의 체계를 부단히 갱신하려고 한다. http://www.kde.org 에 있는 KDE 홈페지로부터 임의의 판본의 Linux 에서 쓸수 있는 KDE 를 갱신할수 있다. 최신판 KDE 파일들은 또한 RPM 형식으로 리용할수도 있는데 이것들은 http;//www.rpmfind.net 에 적재되여 있는 많은 배포물을 위한것이다.

최신판프로그람묶음을 내리적재하면 htdig,libmng,qt 로 시작하는 여러가지 프로그람묶음을 비롯한 《기본》프로그람묶음은 물론 앞의 목록에서 명명된 파일들의 최신판본들을 찾을수 있다. 앞의 장들에서 서술된것처럼 일단 필요한 프로그람묶음을 내리적재하였다면 다음의 지령가운데서 임의의 하나를 쓸수 있다. 여기서 -i 는 처음에 프로그람묶음을 설치하는것을, -U 는 자기의 체계를 최신판본프로그람제품으로 갱신한다는것을 의미한다.

```
# rpm -i filename
# rpm -U filename
```

일러두기 X Windows 에서 RPM 프로그람묶음을 위해서 Kpackage 혹은 GnoRPM 편의프로그람을 리용할수 있다. 설치가 끝나면 Kpackage 는 System → Package Manager 에 있는 KDE 기본차림표로부터, GnoRPM 은 GNOME Programs->System->GnoRPM 에 있는 KDE 의 기본차림표로부터 선택할수 있다.

KDE 의 기동

Red Hat Linux 7.1 에서 기정탁상환경은 GNOME 이다. 일단 설치 CD 나 인터네트에서 내리적재하여 KDE 를 설치했다면 기정 GNOME 도형처리가입창문으로부터 KDE 를 기동할수 있다. 이를 위하여 다음의 단계들을 거친다.

1. Session 차림표를 선택한다.
2. 가능한 탁상들의 목록에서 KDE 를 선택한다.
3. 사용자이름과 통과암호를 주어서 가입한다.

KDE 를 기동하면 그림 11-1 과 같이 초기화면이 나타난다.

보통 조종탁가입재촉문으로 가입해서 startx 지령으로 X Windows 를 기동하면

startx 지령이 실행되기 시작할 때 KDE 가 기동하도록 체계를 구성해야 한다.

이것을 고려하면 다음의 세개의 선택항목을 리용할수 있다.

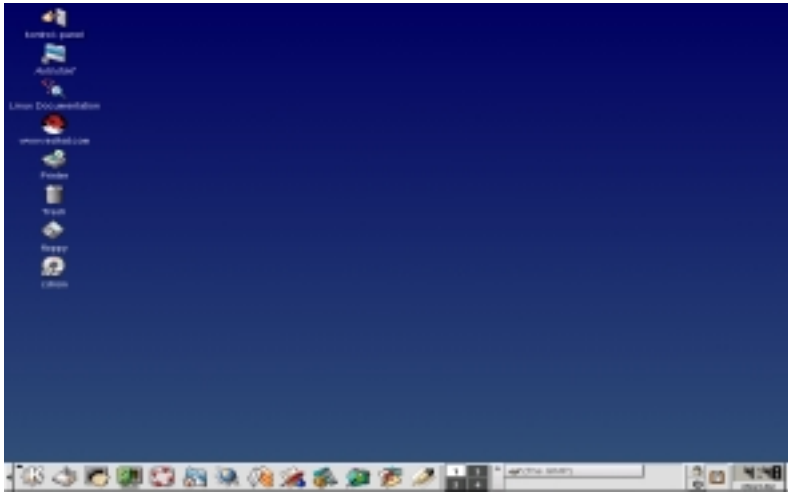



그림 11-1. 여러가지 탁상아이콘이 있는 초기 KDE 화면

- 모든 사용자가 리용하는 기정탁상을 바꾸려면 /etc/x11/xinit/Xclients 에서 볼수 있는 전체 Xclients 를 편집할수 있다.
- 자기의 기정탁상만을 바꾸려면 전체 Xclients 파일을 자기의 홈등록부에 복사하고 거기에서 편집해야 한다. 다음의 지령으로 그 파일을 복사할수 있다.

$ cp /etc/X11/xinit/Xclients ~/.Xclients

첫 두 경우중 어느 한 경우에 그 파일을 한번만 바꿀수 있다. 이 파일의 다섯번째 행은 처음에 다음과 같이 시작된다.

```
PREFERRED=
```

이 행을 다음과 같이 바꿀수 있다.

```
PREFERRED=startkde
```

일단 그것을 바꾸고 보존하면 X Windows 는 startx 지령으로 KDE 탁상을 기동할수 있다. 전체 Xclients 파일을 바꾸면 모든 사용자는 KDE 를 자기의 기정탁상으로 가지게 된다.

- 자기의 기정탁상만을 바꾸려면 Red Hat Linux 7.1 의 Switchdesk 지령을 주어야 한다. 이 지령을 실행하면 탁상환경을 바꿀수 있는 Desktop Switcher 대화칸이 펼쳐 진다. 실례로 KDE 를 선택하면 자기의 홈등록부에 있는 .Xclients-default 파일을 한개 행으로 수정한다. 즉

    exec startkde

KDE 의 끝내기

KDE 를 끝내려면 Logout 를 기본차림표에서 선택해야 한다. 화면이 흐려 지고 End KDE Session 확인창문이 열린다. 여기서 Logout 를 선택하면 KDE 가 끝난다.

Logout 를 선택할 때 어떤 창문들이 열려 있고 다음번에 KDE 를 기동할 때 이러한

창문들을 재생하려면 Restore Session When logging In Next Time 검색칸을 선택해야 한다. 다음번에 KDE 를 기동시키면 KDE 나 기타 도형처리응용프로그람이 실행되기 시작하고 창문이 다시 열리므로 Logout 를 선택했을 때와 같은 탁상을 재생할수 있다. 모든 응용프로그람들이 성공적으로 재실행될수 없다고 해도 KDE 는 그렇게 하려고 한다.

주의	보통 X Windows 체계를 끝내는 기정건반방식을 리용하여 KDE 를 끝낼수 있다. 즉 Ctrl+Alt+Backspace 건을 누르면 된다. 그러나 이 방법은 비상시(도형처리가 잠그어 진 경우)에만 리용하여야 한다. 왜냐하면 KDE 를 끝내기 위해서 이 건반을 리용하면 일부 자료를 잃을수 있기때문이다.

Logout 를 선택했을 때 펼쳐 져 있던 창문들에 대한 정보를 비롯하여 KDE 환경상태들은 홈등록부에 있는 한개이상의 숨은 파일에 모두 기억되는데 그것들에는 .kde/share/config 보조등록부안의 .kderc, .gtkrc-kde 파일들이 포함된다. 필요하다면 이 파일들을 다시 조사할수도 있지만 다음번에 그것을 실행시킬 때 KDE 가 혼돈되지 않도록 이 파일들에 대한 변경을 심사숙고하여야 한다.

주의	파일관리기와 망편의프로그람과 같은 특수응용프로그람상태에 대한 정보는 응용프로그람의 구성파일에 보관되며 일반적으로 **.kde/share/config** 아래의 홈등록부에 놓인다.

도형방식가입을 위한 kdm의 리용

Linux 체계에 도형방식가입을 위한 xdm 혹은 GNOME 표시관리기(gdm)를 리용하는데 습관되여 있다면 KDE 와 함께 제공되는 도형방식가입응용프로그람인 kdm 도 리용해 보고 싶을것이다.

gdm 과 kdm 의 기능은 아주 류사하다. kdm 은 KDE 조종쎈터(이 장의 마지막절 《 KDE 탁상의 탐색》에서 보게 되는)를 리용하여 도형처리형식으로 구성할수 있다.

도형방식가입을 위한 구성

Red Hat Linux 7.1 도형방식가입을 가능하게 하려면 우선 X Windows 체계가 돌아가는가를 확인하고 다음 **/etc/inittab** 파일에서

id:3:initdefault

으로부터

id:5:initdefault

로 변화시킴으로써 실행준위로 3이 아니라 5를 리용한다.

일단 KDE 가 설치되였으면 xm 이나 gdm 을 kdm 으로 바꾸는것은 간단하다. 본문편집기에서도 /etc/inittab 파일을 열고 다음과 같은 행을 배치한다. 그것은 도형방식(x)가입을 리용하는데 무슨 프로그람이 실행되여야 하는가를 나타낸다.

```
x:5:respawn:/etc/X11/prefdm -nodaemon
```

/etc/X11/prefdm -nodaemon 에서 다른 탁상관리기도 볼수 있다. 이 행을 변화시키면 다음과 같다.

```
x:5:respawn:/usr/bin/kdm -nodaemon
```

이제는 체계가 재기동하면 kdm 도형방식가입을 리용할수 있다.

kdm 의 구성

kdm 을 리용하면 인사말문자렬, 대화칸의 아이콘 및 현시되는 배경색 혹은 벽지도형 등을 비롯하여 많은 도형방식가입특성을 구성할수 있다.

이미 앞에서 본 /etc/inittab 에 변화를 만들었다고 하면 다음과 같은 단계들로 kdm 을 구성할수 있다.

1. 도형방식가입대화칸을 리용하여 Linux 체계에 가입한다.
2. KDE 화면의 왼쪽아래구석에 있는 K 아이콘을 찰칵하여 기본차림표를 연다.
3. Preferences → System → Login Manager 를 선택한다.

kdm 구성대화칸은 그림 11-2 와 같다.

그림 11-2. Kdm 도형처리가입화면의 특성을 구성하기 위한 Login Manager 대화칸

주의	뿌리사용자로 KDE 에 가입하지 않았으면 뿌리통과암호에 대한 재촉문이 나타난다.

Appearance 표쪽(그림 11-2 에서 보여 준)우에서 다음과 같은 항목들을 선택할수 있다.

- 도형방식가입창문에서 리용되는 인사말(greeting)
- 가입창문안에서 쓰이는 등록표식(logo) 이것은 실지 등록표식 혹은 시계(clock)

로 될수 있다. Show Logo 를 선택하면 KDE 등록표식단추를 찰칵해서 리용할 수 있는 등록표식을 열람할수 있다.

- KDE, Motif
- SGI 즉 Windows 와 같은 GUI 형태
- 도형방식가입창문에서 통보문들에 쓰이는 언어 및 통용어

Font 표쪽은 Greeting, Fail, Standard 통보문들에리 리용되는 본문서체를 설정한다. 서체를 변경시키려면 내림목록에서 필요한 서체를 선택하고 Change Font 단추를 찰칵해야 한다. 그러면 Select Font 대화칸이 나타나며 거기에서 필요한 서체를 선택할수 있다.

Background 표쪽(그림 11-3 에서 보여 준)으로부터 가입화면에 대한 배경색패턴 혹은 배경도형을 선택할수 있다.

선택된 색배경이 마음에 들면 다음의것들을 포괄하고 있는 Mode 내림칸을 찰칵하여 여러가지를 선택할수 있다.

- 단일한 색으로 된 판(두개의 색띠가운데서 첫번째 색띠)
- 한쪽에서 다른쪽으로 두 색을 혼합시키는 수평그라디엔트
- 우에서 아래로 두 색을 혼합시키는 수직그라디엔트(기정방식)

색을 선택하려면 Color1 혹은 Color2 를 찰칵해야 한다. 그러면 Select Color 대화칸(그림 11-4 와 같은)이 현시되며 거기에서 요구하는 색을 선택할수 있다.

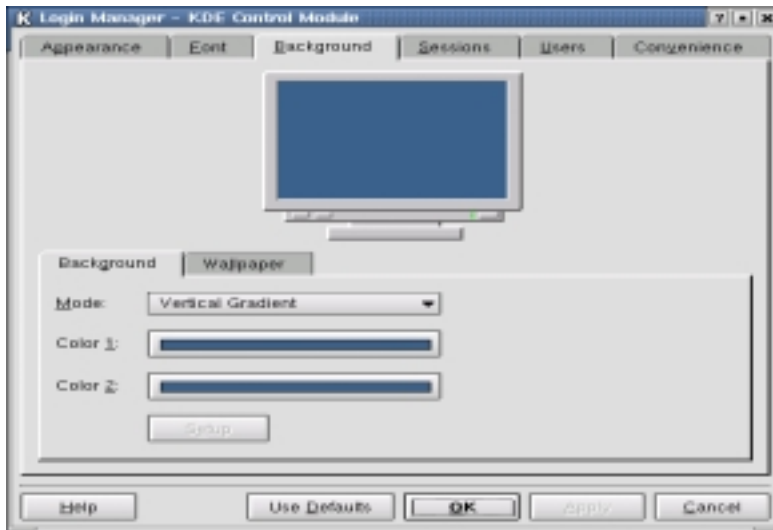



그림 11-3. Background 표쪽으로부터 kde 가입창문의 뒤에서 요구하려는 색갈이나 벽지의 선택

가입화면의 배경에 벽지도형방식을 리용하려면 현시장치의 그림아래에 있는 Wallpaper 표쪽을 선택해야 한다.

KDE 2.1.1 과 함께 25 이상의 표준벽지도형들이 제공된다. Wallpaper 내림목록에서 임의의것을 선택할수 있다.

벽지로 리용하려는 또 다른 도형이 있으면 Browsing 단추를 찰칵하고 도형파일을

보관한 등록부에서 그것을 선택해야 한다.

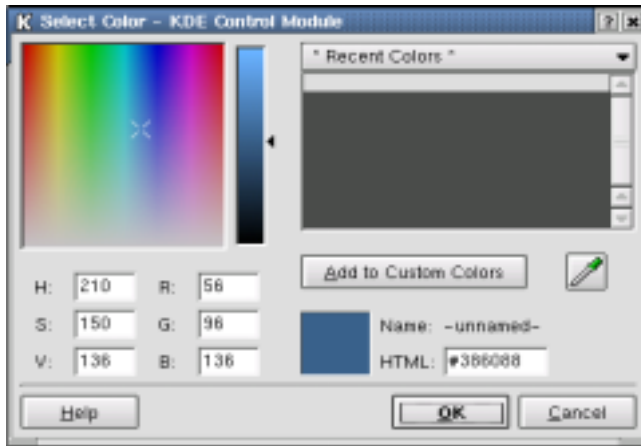



그림 11-4. KDE Select Color 대화칸에서 kdm 배경색의 설정

일러두기 KDE 벽지로 지정한 파일이 삭제되지 않도록 표준 KDE 벽지등록부 /usr/share/wallpapers 에 벽지로 리용하려는 도형파일을 복사하는것이 좋다.

Mode 선택내림목록에 있는 단추들을 선택하여 배경의 각이한 위치들에 벽지도형을 배치할수 있다. 선택항목들은 다음과 같다.

- Tiled 는 전체 배경에 벽지도형을 반복적인 패턴으로 리용하기 위한것이다.
- Centered 는 kdm 화면의 중간에 한개의 벽지를 배치한다. 이 경우에 주로 화면에 kdm 가입창문이 놓일수도 있다.
- Scaled 는 화면구석까지 벽지를 확장한다.

주의 Tiled 나 Scaled 와 같은 벽지선택항목을 리용하면 선택한 배경색은 아무것도 보이지 않는다. 벽지도형없이 배경색만을 리용하려면 Wallpaper 내림목록에서 No Wallpaper 항목을 선택해야 한다.

Sessions 표쪽은 가입대화조종(session)이 어떻게 정의되는가를 지정한다. 실례로 CDE 와 같이 설치된 기타 창문관리기들을 추가할수 있다. Users 표쪽은 가입창문안에 사용자들이 아이콘으로 보여 지도록 구성한다. 마지막으로 Convenience 표쪽을 리용해서 KDE 는 가입에 대한 통과암호를 요구하지 않는 상태를 설정할수 있다.

필요한 kdm 선택항목들을 결정했으면 대화칸을 닫고 설정한것들을 기억시키기 위해서 OK 를 찰칵한다. 그러면 KDE 가입화면에서 그것들을 볼수 있다.

기타 KDE 응용프로그람의 추가

KDE 를 설치하면 이 장의 마감에서 설명하는바와 같이 편의프로그람들과 오락프로그람들의 종합적인 묶음을 가지게 된다. 그러나 많은 응용프로그람들은 오직 설치요구를 만족시키는데 쓰일수 있다.

개별적으로 흥미 있는 어떤 KDE 응용프로그람을 새로운 판본으로 갱신하거나 기정

KDE 배포물에 없는 임의의 응용프로그람을 내리적재할수도 있다.

KDE 응용프로그람들의 기본원천은 http://apps.kde.com의 KDE 응용프로그람 Web 싸이트에 있다. KDE 응용프로그람 Web 싸이트는 정기적으로 갱신되며 여기에는 800 개이상의 KDE 응용프로그람들이 들어 있다. 이 개개의 프로그람들은 간단히 설명되고 FTP 내리적재등록부에 직접 련결된다.

다음의것들은 이 응용프로그람페지에 볼수 있다.

- KDE 사무처리프로그람묶음(Office)
- BIND(DNS)를 구성하기 위한 편의프로그람
- 아이콘과 색배렬의 추가적인 설정(KDE 주제)
- 비데오파일재생기
- KDE Telnet 응용프로그람
- ISDN 선관리도구
- 무릎형콤퓨터전원관리편의프로그람
- 사용자관리도구
- 수십가지의 오락프로그람

이것은 KDE 체계에 이미 설치되여 있는 KDE 응용프로그람에는 없다. 다행히 이러한 대부분의 응용프로그람들은 이 장과 앞의 장들에서 본것처럼 설치할수 있는 정규적인 RPM 프로그람묶음으로 리용할수 있다.

그러나 새로운 KDE 사용자에 관해서 일부 추가적인 KDE 응용프로그람들은 원천코드로 배포된다. 그것들을 리용하자면 개발프로그람묶음을 설치하고 Linux 체계에서 원천코드를 콤파일해야 한다.

이것은 실제로는 그다지 어렵지 않다. 그 문제들은 개발도구를 잘 모르는 사용자들에게만 생기는것이다. 이 경우에 나타나는 오유나 경고, 통보문을 해결하는데 많은 시간이 소비될수 있다.

일단 KDE, Qt, X Windows 체계에 설치된 개발서고가 있으면 다음의 간단한 지령들을 리용하여 내리적재하고 풀어 놓은 원천코드계층구조를 준비할수 있다. 즉

```
# configure
# make all
# make install
```

이 지령에 의하여 새로운 응용프로그람들은 콤파일되여 곧 실행할수 있게 KDE 차림표들이나 2 진코드등록부들에 배치된다. 일부 응용프로그람들은 그것들을 콤파일하고 설치하는데 더 복잡한 단계들을 요구할수 있다. 이러한 경우에 세부화된 지령들을 제공하기 위하여 거의 모든 응용프로그람들에 들어 있는 README 파일과 설치파일을 고찰해보아야 한다.

주의 원천코드로부터 KDE 최신판본을 콤파일하는 상세한 지령을 보려면 http://www.kde.org/install-source.html를 조종해야 한다.

KDE 탁상의 탐색

일단 KDE 가 설치되고 실행할수 있는 환경이면 그 리용법을 배우기 시작할수 있다. KDE 탁상에 있는 요소들은 대체로 잘 알고 있는것들이다. 그림 11-5 는 기본차림표가 펼쳐 진 빈 KDE 탁상을 보여 준다. 다만 탁상의 screenshot 를 제공하기 위하여 The

GIMP 가 열려 져 있을뿐이다(과제띠를 보시오.).

다음과 같은 요소들이 탁상에 나타난다.

- 배경은 쉽게 호출할수 있는 프로그람과 파일아이콘을 보여 준다(8 개의 아이콘들이 기정으로 표시된다.).
- 조종판은 화면의 아래에 가로 놓인 아이콘띠이다(조종판의 위치와 조종판에 있는 아이콘들을 다시 구성할수도 있다.). 아이콘을 찰칵하면 일반적으로 리용되고 있는 프로그람을 개시한다.
- 과제띠는 KDE 에 펼쳐 진 개개의 기본응용프로그람창문에 대한 단추를 보여 준다. 단추를 찰칵하면 그 응용프로그람의 창문이 전개된다.
- 기본차림표는 조종판에 있는 K 아이콘을 찰칵하여 볼수 있다. KDE 의 모든 주요기능들은 기본차림표와 그의 보조차림표에서 리용할수 있다.

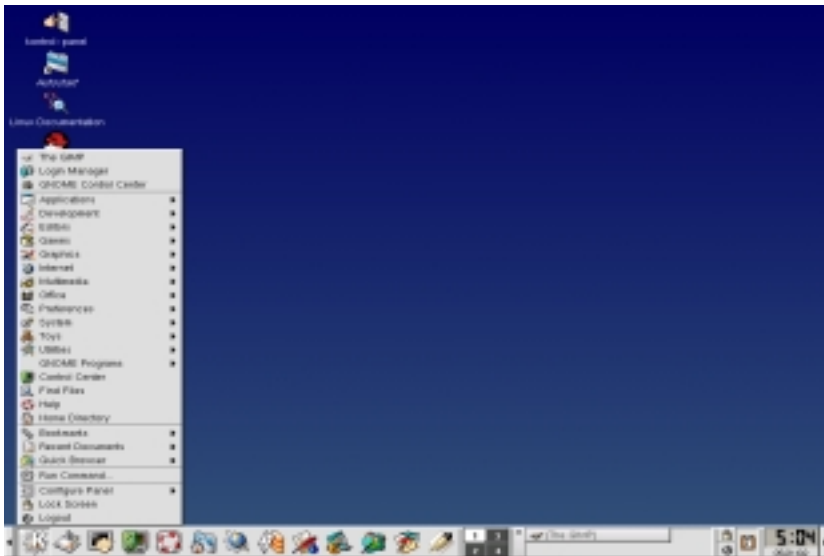



그림 11-5. 초기 KDE 탁상: 여기서는 기본차림표가 열리고 아래에 조종판이 놓이며 초기탁상아이콘을 쉽게 알아 볼수 있다.

기본차림표에서 항목을 선택하거나 조종판의 단추들을 찰칵하여 KDE 의 응용프로그람들을 실행할수 있다.

KDE 응용프로그람의 기동

KDE 도형처리환경에 대해서 경험 삼아 두개의 KDE 응용프로그람들을 보기로 하자. KDE 본문편집기와 말단모방기창문은 좋은 실례이다. 지령행창문을 펼치기 위하여 기본차림표에서 System→Terminal 을 선택한다. 다음 도형방식 KDE 본문편집기를 열기 위하여 Editors→Text Editor 를 선택한다.

일러두기 일단 말단모방기창문이 펼쳐 지면 차림표에서 KDE 프로그람이름을 선택하거나 조종판아이콘을 찰칵하지 않고도 임의의 프로그람이름을 입력할수 있다. 실례로 지령행에 **kedit &**나 **/usr/bin/kedit &**지령을 입력하여 본문편집기를 기동시킬수 있다.

조종판으로부터 옆의 그림과 같은 Terminal Shell 단추를 찰칵하여 다른 말단창문을 기동시킬수 있다.

KDE 탁상을 실행하고 있으면 KDE 응용프로그람은 서로 호상작용할수 있고 체계자원들을 보존하기 위하여 일반도형처리서고들을 리용하며 같은 감각을 가지기때문에 대부분의 다른 X Windows 체계도형방식응용프로그람과 특수하게 대조된다.

물론 다른 도형방식이나 본문방식응용프로그람들도 KDE 에서 리용할수 있다. GNOME 에 기초한 응용프로그람이 들어 있는 일반적인 도형방식응용프로그람가운데서 그 일부는 Non- KDE Applications 보조차림표의 KDE 기본차림표에 있을수 있다. Red Hat Linux7.1 환경에서 대부분의 GNOME 응용프로그람은 GNOME Programs 보조차림표에 있다.

GNOME 이나 CDE 와 같은 다른 창문관리기에서와 같이 KDE 에서의 지령행말단창문에서 자기의 기호에 맞는 응용프로그람을 기동시킬수 있다.

이를 위하여 KDE 말단모방기창문에 xv, gimp, gs 와 같은 지령을 줄수 있다(선택된 프로그람이름은 설치에 따른다.).

일러두기 프로그람을 기동시키는 지령을 입력하는 한가지 빠른 방법은 KDE 의 pop-up 지령창문을 리용하는것이다. 이것을 리용하려면 Alt+F2 를 누르고 나타나는 작은 pop-up 창문에 지령을 입력해야 한다.

다중탁상면의 리용

기타 Linux 창문관리기에 대해서 잘 알면 개개가 많은 열린 창문을 포함하고 있는 다중가상탁상을 관리하는 Pager 도 리용할수 있다. 한 가상탁상에 열람기를, 다른 탁상에 문서편집기를, 또 다른 탁상에 코드를 편집하는 말단창문을 가질수 있다.

KDE 는 다중가상작업공간을 제공하기 위해 간단한 개념을 도입한다. 기정으로 KDE 에는 4 개의 가상탁상이 있으며 조종판의 탁상이름이나 번호를 찰칵하거나 탁상 1 에 대해서는 Ctrl+F1 을, 탁상 2 에 대해서는 Ctrl+F2 를 누르는 식으로 4 번째 탁상사이에서 절환할수 있다.

일러두기 탁상들의 번호나 이름을 변경시킬수도 있다. KDE 기본차림표로부터 Preferences Look & Feel Desktop 를 선택한다. 제시되는 대화칸에서 Number of Desktop 표쪽을 찰칵한다. 이 창문우에 있는 Slider 는 16 개의 가상탁상가운데서 임의의것을 구성한다. 가상탁상과 관련된 개개의 본문창문에서 탁상이름을 수정할수 있다.

Ctrl 건을 계속 누른 상태에서 Tab 혹은 Shift+Tab 를 반복하여 눌러서 탁상사이를 순환할수 있다.

응용프로그람창문에 의한 작업

KDE 에서 응용프로그람창문에 의한 작업은 이미 본 다른 도형처리환경에 의한 작업과 류사하다. 창문들을 옮기고 크기를 변화시키고 닫는것과 같은 방법들이 KDE 에서도 리용될수 있다.

KDE 탁상에서 여러개의 창문을 열었을 때 그 창문의 임의의 곳에 찰칵하여 임의의 창문을 동작시킬수 있다.

일러두기 KDE focus policy 는 어느 창문이 KDE 에서 능동창문인가를 결정한다. 창문에 지시자를 가져다 놓는것과 같은 초점잡이에 익숙되였다면 Preference→Look&Feel→Windows Behavior 를 선택하여 초점방향을 바꿀수 있다. 제시된 대화칸에서 Actions 표쪽을 찰칵하고 요구하는 초점방향을 선택한다.

KDE 의 다중가상탁상에는 동시에 열린 많은 창문에서 쉽게 작업할수 있을뿐아니라 열린 응용프로그람창문사이를 절환할 때 리용할수 있는 여러가지 기능들도 있다.

- 과제띠는 펼쳐 진 매 창문의 단추를 현시한다. 과제띠단추를 찰칵하여 그 창문을 능동창문으로 만들고 그것을 정면에 보이게 할수 있다.
- Microsoft 창문들과 마찬가지로 열려 진 창문사이에서 초점을 절환하려면 Alt 건을 누른 상태에서 Tab 나 Shift+Tab 건을 반복하여 눌러야 한다.
- KDE 대화조종에서 창문과 탁상목록을 열려면 마우스의 중간단추를 눌러야 한다. 이 목록에서 해당한 창문을 선택하여 능동으로 만들거나 정면에 보이게 할수 있다.

주의 두단추마우스인 경우 제 4 장 《Red Hat Linux 7.1 의 설치》를 보면서 마우스중간단추모방을 구성할수 있다. 중간단추찰칵을 모방하려면 왼쪽과 오른쪽단추를 동시에 찰칵해야 한다. Windows List 를 호출하는 《단추》를 변경시킬수도 있다. Red Hat Linux 7.1 에서 /usr/sbin/mouseconfig 프로그람을 리용하여 마우스중간단추를 설정할수도 있다.

파일체계의 열람

도형방식체계리용에서 큰 몫을 차지하는것은 콤퓨터에서 파일들을 호출하는것이다. Linux 는 이 기능을 실행하는 완전히 집적화된 파일관리기를 제공한다. QuickBrowser 를 리용하여 KDE 기본차림표로부터 직접 파일체계를 호출할수 있지만 이 수법은 파일관리기창문의 동작기능을 제공하지 않는다.

이 절에서는 KDE 파일관리기창문과 QuickBrowser 차림표를 가지고 작업하는 방법을 고찰한다.

Konqueror Windows 의 리용

KDE 에서 파일관리기는 Konqueror 라고 하며 이것은 Web 열람기로도 쓰인다. 조종판에 있는 아이콘을 찰칵하여 Konqueror 을 기동할수 있다.

이것은 그림 11-6 에서 보는것처럼 Web 열람기와 같이 Konqueror 을 기동한다.

그림 11-6. Konqueror

한편 열에 보여 준 조종판에 있는 Home Directory 아이콘을 찰칵하여 Konqueror 의 홈등록부를 열수 있다.

Konqueror 창문은 그림 11-7 과 같이 현시된다.

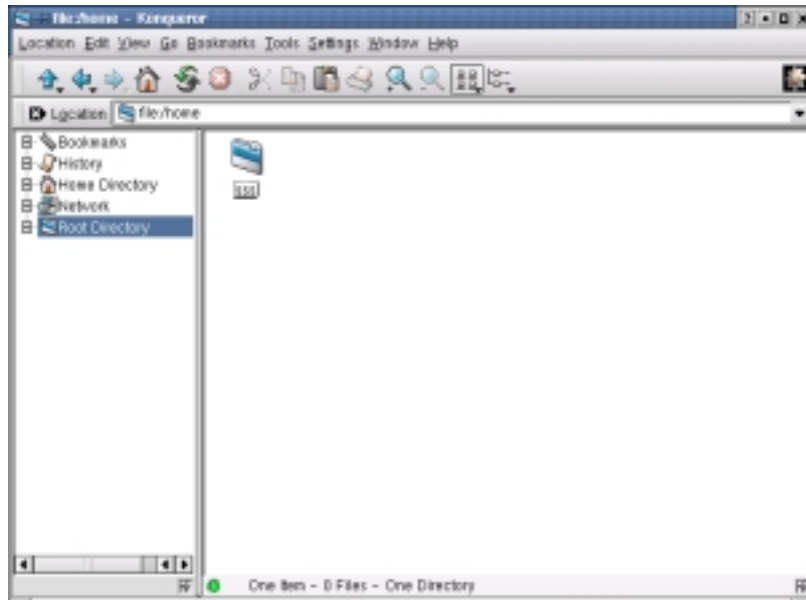



그림 11-7. Konqueror 에 의한 홈등록부내용과 기타등록부의 열람

기정인 파일체계현시는 오른쪽조종판에 있는 중간크기의 아이콘을 리용한다(파일형태에 따라 KDE 가 지정). Konqueror 창문꽉대기에서 도구띠와 Location 행을 볼수 있다. 또한 Microsoft Windows 의 Explorer 에서 Folders 판과 같이 왼쪽판에 등록부나

무도 있다.

더 상세한 정보를 보기 위하여 이 파일들의 현시를 변경시키는것이 쉽다. 즉 Konqueror 도구띠에서 MultiColumn 혹은 Text View 와 같이 View-> ViewMode 로부터 또 다른 표시항목을 선택한다. 또한 홈등록부에 있는 전체파일을 보려면 View 차림표에서 Show Hidden Files 항목을 선택할수 있다. 도구띠의 왼쪽끝에 있는 웃방향화살아이콘을 찰칵하면 어떤 등록부로 넘어 간다.

임의의 등록부를 찰칵하면 그 등록부가 열리고 같은 Konqueror 창문안에 그 등록부내용들이 현시된다. 새로운 Konqueror 창문을 현재 위치에서 열려면 Location→New Windows 를 선택하든가 Ctrl+N 을 눌러야 한다.

파일이나 등록부를 옮기거나 복사하려면 해당한 아이콘을 찰칵하고 끌어서 그것을 옮기거나 복사해 넣으려는 등록부에 놓아야 한다(튀여나오기차림표에서 More 나 Link, Copy 를 선택할수 있다).

Konqueror 에서 한개의 아이콘을 찰칵하면 KDE 는 그 아이콘을 기동하려고 한다. 그것이 본문파일이라면 KDE 는 KDE 본문편집기의 파일을 연다. 도형파일은 도형현시도구로 현시된다. 응용프로그람들은 될수록 Konqueror 의 오른쪽조종판에 있는 파일을 연다.

KDE 가 그 파일을 열수 있는 도구를 찾을수 없으면 Open With 대화칸이 현시되며 그림 11-8 에서 보여준것처럼 찰칵한 그 아이콘을 기동할 때 리용할수 있는 프로그람을 설정해야 한다. 이 대화칸에는 KDE 기본차림표의 Known Applications 차림표가 들어 있다.

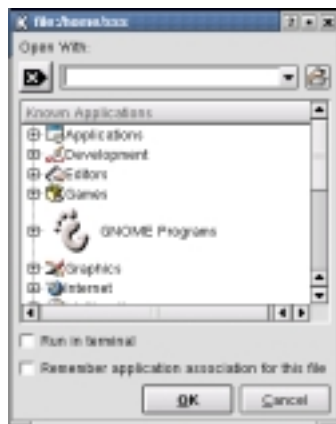
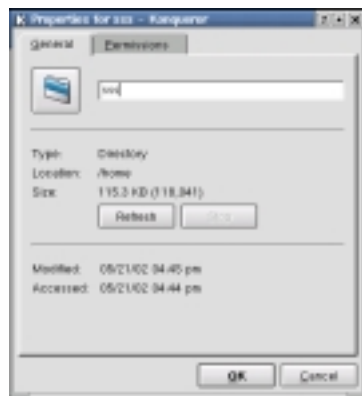



그림 11-8. 파일을 열기 위한 응용프로그람의 선택

파일속성의 변경

Konqueror 에서 호출한 파일이나 등록부의 속성들을 바꿀수 있다. 등록부나 파일속성들을 현시하려면 파일 혹은 등록부아이콘을 오른쪽찰칵해야 한다. 제시되는 튀여나오기차림표에서 Properties 를 선택한다. 그림 11-9 에서 보는것처럼 Properties 대화칸에는 적어도 그 파일 혹은 등록부와 관계되는 이름과 허가들을 변경할 때 리용할수 있는 표쪽으로 나타난다.

그림 11-9. Properties 대화칸에 의한 Konqueror 창문에 대한 정보의 현시와 변경

주의 Properties 대화칸에 나타나는 특수한 표쪽들은 현시하고 있는 대상의 형태에 따라 달라 진다. 실례로 KDE Link 파일과 관련되는 속성에는 련관된 편의프로그람과 응용프로그람이 포함된다.

Edit→Create→New 보조차림표를 리용하여 새로운 대상을 만들수 있다. 이 보조차림표에서 KDE 가 체계자원들을 호출하는데 리용할수 있는 새로운 폴더(보조등록부)나 HTML 파일들, 응용프로그람련결(links), 삭제가능한 구동장치들과 같은 새로운 정보대상들을 만들수 있다.

Konqueror 에 의한 열람

Konqueror 는 Web 열람기창문과 비슷하며 실제로 Web 페지들을 호출하는데 리용할수 있다. Index.html 파일을 포함하고 있는 등록부에서 국부적인 파일체계를 열람한다면 등록부의 내용에 대한 아이콘보다 오히려 그 파일을 현시하여야 한다.

일러두기	그 등록부의 HTML 색인대신에 등록부에 있는 파일을 보려고 View→Use Index.html 을 선택하지 말아야 한다.

어떤 등록부가 현시되여 있으면 Location 마당을 찰칵하고

```
http://www.sybex.com
```

혹은

```
ftp://ftp.kde.org
```

와 같이 완전한 Web URL 을 입력할수 있다. 현재 인터네트에 접속되여 있으면 적합한 인터네트페지를 적재하고 Konqueror 에 현시한다.

Konqueror 의 구성

Web 열람기와 파일관리기로서 Konqueror 의 작용과 동작을 구성할수 있다. Konqueror 도구띠에서 Setting→ Configure Konqueror 를 선택한다. 그림 11-10 과 같이 9 가지 부류로 Konqueror 를 구성할수 있다.

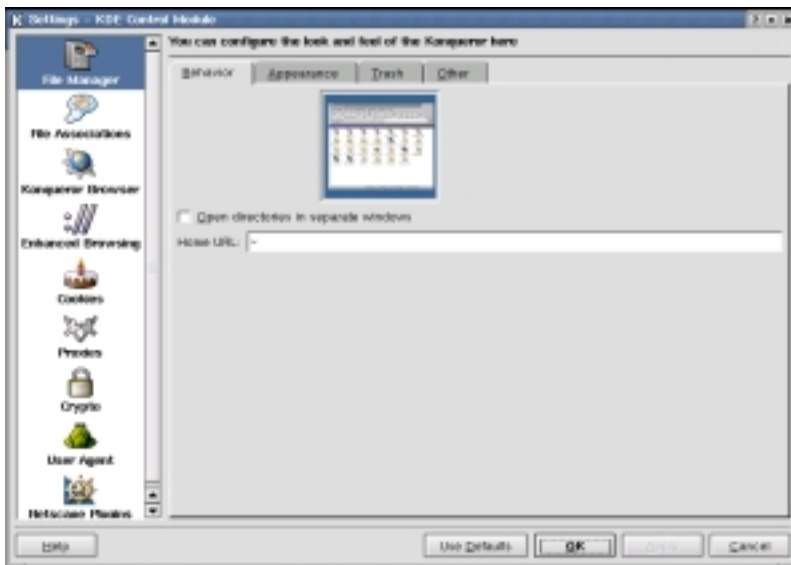



그림 11-10. konqueror 구성창문

- File Manager 는 파일관리기창문의 동작과 외관을 설정한다.
- File Association 은 KSpread 와 같은 전용응용프로그람에 XLS 파일과 같은 형태의 파일을 결합시킨다.
- Konqueror Browser 설정은 련결정보, 본문크기, Java, 접속장치들과 같은 규칙들을 관리한다.
- Enhanced Browsing 은 각이한 Web 싸이트에 필요한 여러가지 건반들을 구성한다. 이것은 하나의 열람기로서 Konqueror 에만 적용할수 있다.
- Cookies 는 각이한 Web 싸이트들로부터 입수한 구성통보문을 구성한다.
- Proxies 는 proxy 봉사기콤퓨터로 인터네트나 기타 Web 망들에 접속된다.
- Crypto 설정들은 리용하는 암호화/복호화알고리듬을 설정한다.
- User Agent 는 리용하는 Web 봉사기가 열람기를 조사하는 방법을 조종한다.
- Netscape Plugins 는 여러가지 파일형태들과 관련된 집속모듈들을 제시한다.

고속열람기의 리용

Konqueror 가 파일들을 도형방식으로 관리하는 좋은 수법을 제공하지만 프로그람을 즉시에 기동시키거나 어느 한 자료파일을 호출하려면 Konqueror 가 좀 느리게 보일수 있다. 고속열람기(Quick Browser)는 Konqueror 를 열지 않고 KDE 기본차림표를 리용하여 파일체계의 어느 한 부분을 호출할수 있게 한다.

물론 Quick Brouser 차림표들을 거쳐 등록부들사이에 파일들을 끌어다 놓거나 항목들에 대한 Properties 창문들을 열수 없지만 응용프로그람을 기동시키거나 자료파일을 여는데는 나무랄데 없는 도구이다.

Quick Brouser 를 호출하려면 기본차림표에서 Quick Brouser 를 선택해야 한다. Quick Brouser 보조차림표에는 3 개의 선택항목이 있다. 즉

- Home Directory 선택항목은 홈등록부에서 기동하는 파일들과 등록부들을 호출한다.
- Root Directory 선택항목은 Linux 의 맨 꼭대기준위등록부인 root(/)등록부에서 기동하는 파일들과 등록부들을 호출한다.
- System Configration 선택항목은 /etc 등록부의 구성파일들과 등록부들을 호출한다.

Quick Brouser 가 파일체계를 어떻게 현시하는가를 보려면 기본차림표에서 Quick Brouser→Root 를 선택하고 그림 11-11 에서 보여 준것과 같이 제시되는 보조차림표들을 통하여 순간적인 열람을 진행해야 한다. 다음과 같은 특성들에 주의해야 한다.

- 어느 한 등록부에로 넘어 갈 때마다 그 등록부의 내용들이 현시된다.
- 어느 한 파일을 선택할 때마다 KDE 는 그 파일을 기동하거나 열려고 한다.
- 임의의 보조차림표에서 등록부의 이름을 찰칵하고 Open In File Manager 나 Open In Terminal 을 선택하여 Konqueror 나 그 등록부의 지령행창문을 열수 있다.

실례로 Quick Brouser 차림표에서 Home Directory 를 열람하자. 이를 위해서 홈등록부에서 선택하려는 파일을 찰칵한다. 그러면 KDE 는 련관된 응용프로그람이 그 파일을 연다.

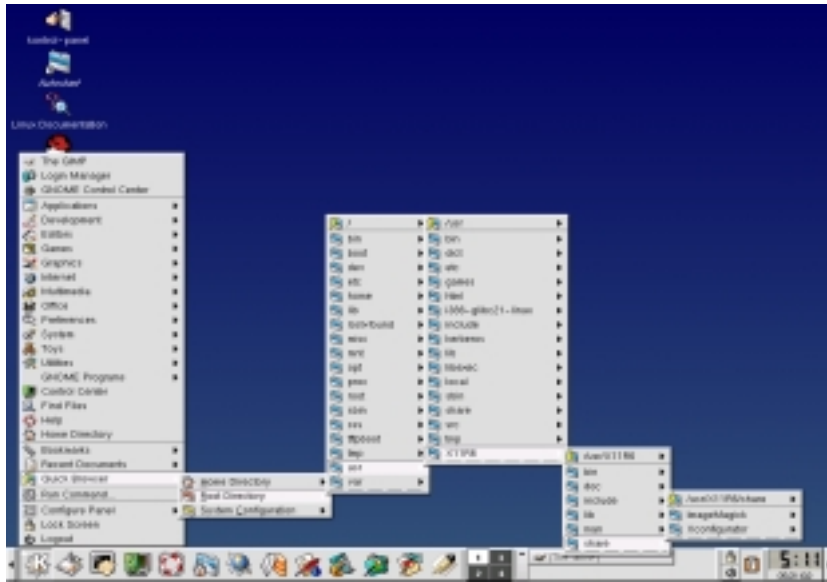



그림 11-11. 파일관리기창문을 열지 않고도 파일체계를 열람할수 있는 Quick Brouser 보조차림표

실례로 Quick Brouser 차림표에서 Home Directory 를 열람하자. 이를 위해서 홈등록부에서 선택하려는 파일을 찰칵한다. 그러면 KDE 는 련관된 응용프로그람이 그 파일을 연다.

KDE 련결파일만들기

이제부터 KDE Link 파일을 고찰하자. 이 파일들은 하나의 체계자원을 설명하는 자그마한 정보를 가진 본문파일이며 따라서 KDE는 그것(체계자원)을 리용할수 있다.

실례로 CD-ROM 구동기에 따르는 KDE 파일은 기정 KDE 탁상부분이다. 이 KDE Link 파일(그리고 그와 관련된 아이콘)은 지령행을 리용하지 않고도 CD-ROM 구동기에서 파일체계를 호출할수 있다. 탁상에서 직접 CD-ROM 내용들을 설치 및 해제하거나 조사할수 있다. 다행히도 KDE에서는 KDE Link 파일을 도형방식으로 만들수 있다.

KDE Link 파일을 실지처럼 보려고 한다면 홈등록부의 탁상보조등록부로 가야 한다. 홈등록부가 /home/mj 라면 /home/mj/desktop 로 가서 본문편집기로 탁상에 있는 아이콘들에 해당한 파일들가운데서 한개의 파일을 열어야 한다.

도형방식의 현시도구인 The GIMP 와 같은 도형처리응용프로그람에 따르는 새로운 KDE Link 파일을 만들려면 다음의 단계를 거쳐야 한다.

1. 홈등록부를 현시하기 위하여 파일관리기(Konqueror)창문을 열고 국부적인 탁상보조등록부에로 넘어 간다.
2. Edit→Create New→Link To Application 을 선택한다. 그러면 Properties 대화칸이 나타난다.
3. 이 경우에 The GIMP 응용프로그람에 맞게 이름의 첫번째 부분을 변경시킨다.

4. 다른 필요한 아이콘의 내용을 보려면 그 아이콘을 찰칵해야 한다. 먼저 그 아이콘을 선택하고 OK 를 찰칵한다.
5. Execute 표쪽(그림 11-12)을 선택한다.
6. Execute 표쪽의 Command 마당에서 이 KDE Link 파일을 리용하여 호출하려는 프로그람을 기동시키는데 필요한 지령을 입력한다.
7. Properties 대화칸을 닫으려면 OK 단추를 찰칵해야 한다.

일러두기 이 응용프로그람에 추가적인 정보를 기록할수 있다. 실례로 General 표쪽에서 새로운 아이콘들을 선택하기 위해서 기정인 아이콘을 찰칵하든가 혹은 이 응용프로그람으로 작업할수 있는 자료형들을 정의하기 위해서 Application 표쪽을 선택한다.

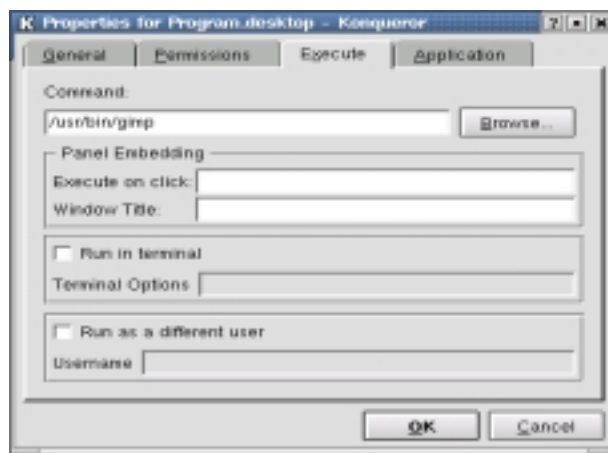



그림 11-12. KDE Link 파일의 Properties 대화칸의 Execute 표쪽으로 응용프로그람을 기동시키는 방법을 정의한다.

이제는 KDELink 파일을 만들수 있다. 다음번에 KDE 에 가입할 때 이 파일에 대한 아이콘이 탁상에 나타날것이다. 여러개의 Link 파일들을 KDE 차림표를 확장하는데 리용할수 있으며 탁상우에 아이콘들을 추가하고 응용프로그람들을 자동실행시키고 KDE 조종판에 아이콘들을 추가할수 있다. 다음항목 《 KDE 의 구성》은 KDELink 파일을 비롯하여 이러한 과제들을 어떻게 완성하는가 하는것을 설명한다.

KDE 의 구성

kdm 도형방식가입프로그람에서 본것처럼 많은 KDE 구성항목들과 KDE 가 제공하는 창문항목들을 이미 보았다. 이 항목들과 기타 많은 설정들은 KDE 조종쎈터로 구성할수 있다. 제일 많이 쓰는 항목들을 여기서 고찰한다.

KDE Control Center 의 리용

KDE 조종쎈터를 기동하려면 기본차림표로부터 KDE 조종쎈터를 선택하든가 또는 기정조종판우의 Control Center 아이콘을 찰칵해야 한다.

Control Center 창문을 열면 창문의 왼쪽에 목록을 보여 주는 보조구역이 나타난다. 이 목록구역에서 임의의것을 찰칵하여 구성표제목록을 열수 있다. 한개의 표제를 찰칵하면 화면의 오른쪽면이 변하면서 구성항목들을 현시한다.

일러두기 Control Center 의 항목들은 기본차림표의 Preferences 보조차림표들과 일치한다. 구성하는 방법에 따라 조종쎈터를 더 편리하게 리용할수도 있다.

주의 Control Center 에서 Apply 를 선택하면 변경한 구성들이 능동화되지만 Control Center 창문을 닫지는 않는다. 닫으려면 File→ Quit 를 선택하거나 제목띠에 있는 Close 단추를 찰칵해야 한다.

Control Center 에는 많은 선택항목들이 있는데 그림 11-13 에서 보는것처럼 11 개 항목들로 갈라 진다. 이 항목들은 보통 KDE 사용자가 가장 많이 리용하는 선택항목에 대한 작업방법을 서술한다.

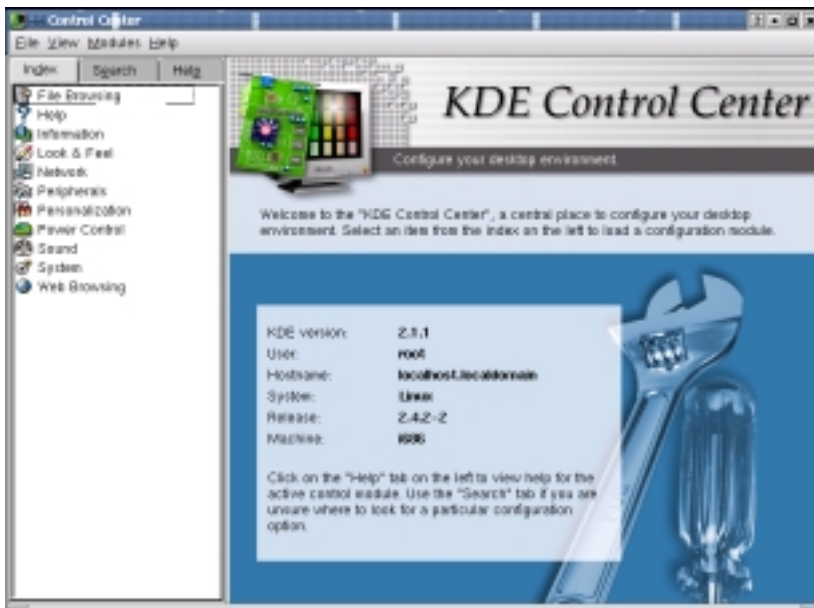



그림 11-13. KDE 조종쎈터

KDE 2.x 에서 변경된 기본선택항목은 차림표띠의 Control Center 왼쪽 웃부분에 있는 Index, Search,Help 표쪽에 있다. Index 표쪽에는 여러가지 구성항목들이 있다. Search 표쪽은 건반에 대한 구성들을 조사한다. Help 표쪽은 현재 강조된 항목에 대한 간단한 개념을 제공한다.

KDE 파일관리

KDEControl Center 의 File Browsing 항목은 파일들의 도형사용자대면부와의 련결

및 조작법을 설정한다. File Associations 보조항목은 파일확장자들을 .txt 및 .jpg 로 만들고 그것들을 전용응용프로그람에 결합시시킨다. KDE 파일관리기인 Konqueror 에서 한개 파일을 찰칵하면 련관된 응용프로그람으로 그것을 연다. 특수한 형태의 파일을 선택할 때 련관되는 응용프로그람을 변화시킬수도 있다.

File Manager 보조항목은 외관, 파일들을 Trash 로 옮길 때의 동작, 련관된 지령행 말단프로그람으로 새 등록부를 여는 Konqueror 의 동작을 구성한다.

Help(도움말)체계

KDEControl Center 는 3 개 층으로 된 도움말체계를 가진다. 고정과 최소화, 최대화단추들에 뒤이어 창문오른쪽구석에는 물음표가 제시된다. 그 물음표를 찰칵하고 해당항목들을 찰칵하면 KDE Control Center 는 그 항목에 대한 간단한 개요를 준다.

조종쎈터의 왼쪽우에 Help 표쪽이 있다. Index 표쪽에서 한개의 항목을 강조하고 그에 해당한 상세한 정보를 보기 위하여 Help 표쪽을 찰칵할수도 있다. 창문마당의 Help 단추를 찰칵하면 Help 표쪽을 선택한것과 같은 효과가 나타난다. 그림 11-14 에서 보여 준것처럼 지정항목에 대한 Help 표쪽을 선택하는 다른 방법은 창문의 아래측면에 있는 Help 단추를 찰칵하는것이다.

마지막으로 Help 정보의 기초에는 《완전한 사용설명서를 보려면 here 를 찰칵하시오.》라는 설명이 있다. 《here》를 찰칵하면 Konqueror 는 KDE Control Center 사용설명서의 해당한 부분을 연다.

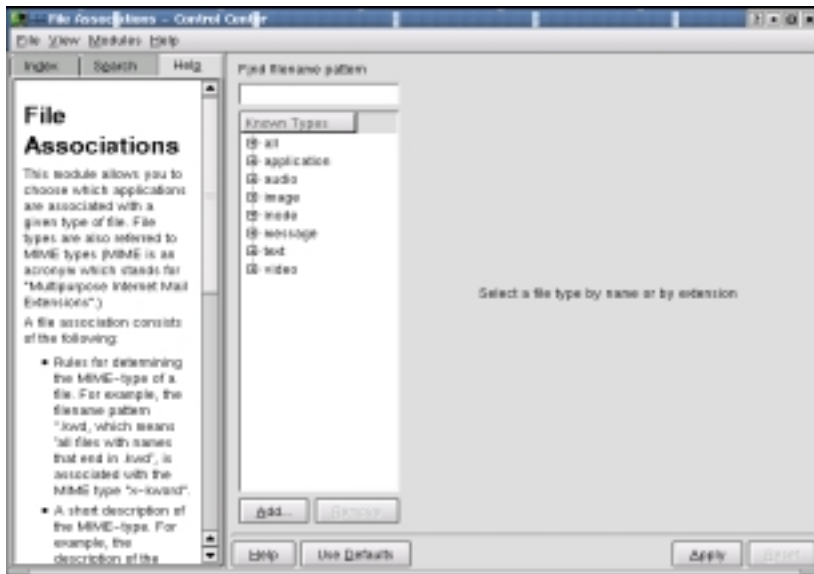



그림 11-14. Help 표쪽에 의한 KDE 조종쎈터의 목적하는 항목에 대한 정보제공

체계정보의 수집

조종쎈터의 Information 항목은 Linux 체계에 대한 상세한 통계와 기술적정보를 제

공한다. 이 표쪽들에 있는 정보는 변경시킬수 없다. 이것들은 개념을 세우는데 효과적이며 체계의 설정과 상태를 알수 있게 한다. 리용할수 있는 보조항목들은 다음과 같다.

Block Devices root 등록부, CD-ROM(/mnt/cdrom), 망련결장치들을 비롯하여 구성설치된 장치블로크들을 제시한다.

DMA-Channels 체계의 DMA 기억호출통로들과 그와 관련된 하드웨어동작을 제시한다.

Devices Linux 가 인식한 장치(그러나 개별적장치는 아니다)들을 제시한다.

IO-Ports 체계에 리용된 I/O 포구주소들을 제시한다.

Interrupts 체계에서 정의된 새치기들과 IRQ 통로에서와 같이 알려 져 있는 새치기들을 제시한다(일부는 지금 시점에서 한개 값을 가질수도 있다.).

KDE IO Slaves nfs, pop3, telnet 와 같은 규약정의프로그람들을 현시한다.

Memory 기억기리용화면을 현시하는데 가상 및 실제체계기억기사용법을 매 초마다 갱신하여 도형으로 제시한다.

PCI 모든 PCI 장치들의 상태에 대한 통보를 제공한다.

PCMCIA 매 꽂기홈에 설치된 PC 카드의 형태를 비롯하여 콤퓨터의 PCMCIA 꽂기홈을 제시한다.

Partitions /etc/fstab 구성파일에서 정의한것처럼 체계에서 리용할수 있는 구획들을 제시한다.

Processor 처리소자의 형, 속도, 기타정보를 비롯하여 처리소자에 대한 정보를 현시한다.

SCSI SCSI 장치들에 대한 정보를 제시한다.

Samba Status Samba 봉사기가 설치되고 실행되고 있다면 현재 가입된 모든 사용자들과 그들이 공유한 Linux 자원들을 제시한다.

Sound Linux 소리체계상태를 제시한다.

X-server X 봉사기에 리용되는 판본, 봉사기, 색계조, 해상도를 현시한다.

KDE 에서 보고느낌의 변경

조종쎈터의 많은 페지들은 KDE 의 현시와 대면부항목들을 지정하는데 리용된다. 이것들가운데서 몇가지만을 고찰하는데 앞에서 본 Login Manager 구성과 같다. 대부분의 흥미 있는 대면부구성항목들은 Look & Feel 항목에서 제공된다.

Background 보조항목은 매 탁상에 대하여 각이한 배경을 설정한다. 이 보조항목에는 그림 11-15 에서와 같은 3 개의 구성표쪽들이 있다. Background 와 Wallpaper 표쪽들은 kdm 가입관리기에 대하여 가능한 배경색과 벽지를 정의한다. 100 개이상의 벽지도형처리가 KDE 에 포함되여 있다. 매개의 탁상은 각이한 벽지와 배경색방안을 가질수 있다. Advanced 표쪽은 배경에 따라 고속기억용량을 제한하며 여러가지 벽지패턴들의 조화를 설정한다.

colors 보조항목에서는 KDE 에서 열리는 모든 창문들의 외관을 정의할수 있다. 이 항목에서 색방안도 선택할수 있다(그림 11-16). 이 보조항목은 Microsoft Windows 의 Display Properties, Appearance 표쪽과 비슷하다. KDE 에서는 또한 추가적인 색방안을 만들거나 매 창문요소(widget 로 불리운다.)를 개별적으로 수정할수 있다.

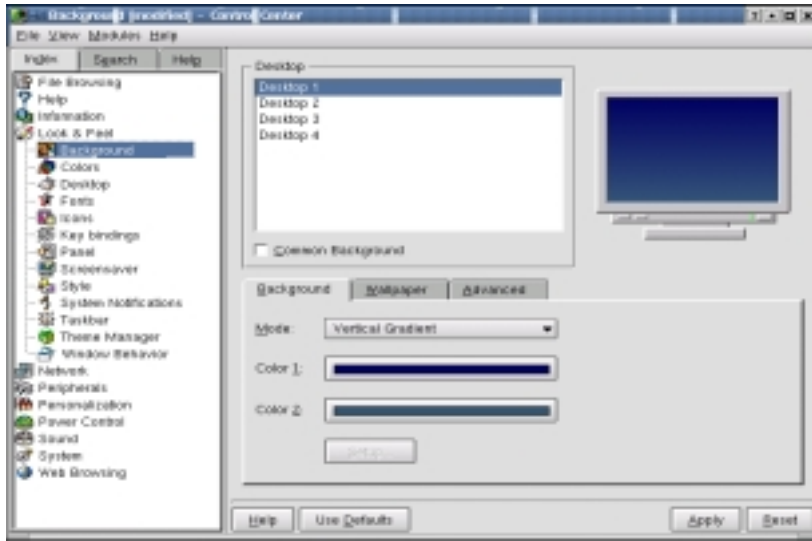



그림 11-15. Background 보조항목에 의한 매 탁상의 배경색과 벽지정의

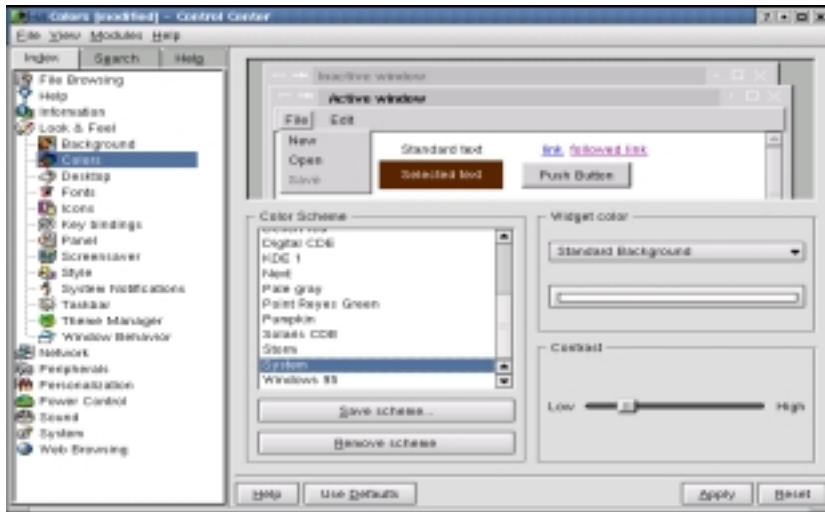



그림 11-16. Colors 보조항목에 의한 모든 KDE 창문의 색방안정의

Desktop 보조항목에서는 탁상의 서체와 아이콘, 능동화된 주위탁상경계선, 매개 마우스단추와 관련된 작용, 여러개의 가상탁상을 정의할수 있다.

조종판보조항목에서는 조종판과 과제띠의 정리를 갱신할수 있다. 이 항목안의 5 개 표쪽들로부터 다음과 같은것들을 구성할수 있다.

- 조종판과 과제띠의 화면위치
- 조종판과 과제띠항목의 동화
- 기본차림표에 있는 특수한 항목
- 지름건단추들의 각이한 부류에 따르는 색갈
- KDE 를 기동할 때 적재되는 애플레트

Screensaver 보조항목(그림 11-17 에서 보여 준다.)에서는 자기의 체계가 얼마간 비능동상태로 있을 때 KDE 가 사용하게 될 화면보호기프로그람을 정의한다. KDE 에는 선택가능한 21 개의 화면보호프로그람들이 있다. 대부분은 Setup 단추를 선택하여 구성할수 있다.

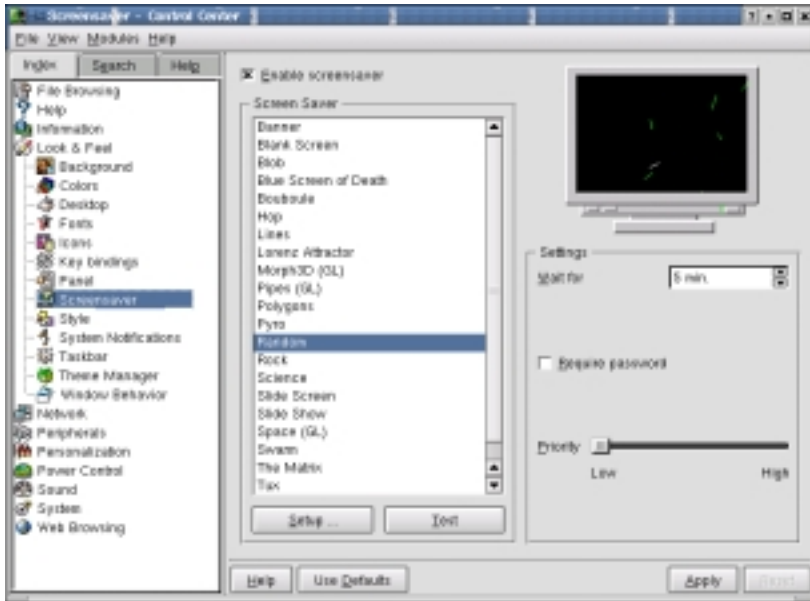



그림 11-17. Screen save 보조항목에 의한 체계에서 리용하는 KDE 화면보호프로그람의 정의 및 구성

일러두기 KDE 기본차림표는 화면을 복귀할 때마다 Linux 통과암호를 요구하는 Lock Screen 항목을 가진다. 콤퓨터에서 벗어 날 때 이 항목들을 리용하기 위하여 기본차림표로부터 Look Screen 을 선택한다.

많은 추가적인 대면부설정들은 조종쎈터의 Windows 항목에서 대화칸을 리용하여 구성할수 있다. 이 항목들은 수준이 높기때문에 몇몇 안되는 사용자들만이 흥미를 가진다. 이것들은 다음과 같은것을 의미한다.

- 각이한 부류의 탁상항목, 제목, 차림표에 대한 서체를 설정한다.
- 각이한 형태의 탁상아이콘을 선택한다.
- 탁상과 독립적인 창문에서 여러가지 기능들에 대한 건반조작을 바꾼다.
- 아이콘과 차림표의 형태와 양식을 구성한다.
- KDE 체계에서의 각이한 작용뿐아니라 KDE 창문관리도구, 긴급통보문들에 대한 통지사건들을 조종한다.
- 능동응용프로그람의 과제띠를 배치한다.
- 탁상의 모든것에 대해서 밑선제목을 설정한다.
- 각이한 마우스작용에 기초한 창문의 초점잡이동작을 조정한다.

망설정에 대한 리해

KDE Control Center 의 Network 항목은 LAN Browsing 과 Talk 구성을 포괄한다. LAN Browsing 은 또한 LISa 로 알려 진 KDE LAN Information Server 도구를 리용하여 Samba 련결을 설정한다. Talk 구성설정들은 Unix 대화데몬과 관련되는데 이것은 현재 긴급통보문프로그람들에 대치되는 지령행방식에 기초하고 있다.

하드웨어설정(주변장치)의 선택

일반적으로 KDE Control Center 로 하드웨어를 구성할수 있다. 그러나 기본하드웨어와 관련된 몇가지 실례들이 있다.

Peripherals 항목에서 건반과 마우스에 대한 정보를 설정할수 있다.

- Keyboard 보조항목은 건반찰칵하기에 대한 자동반복설정과 음량을 정의한다.
- Mouse 보조항목은 왼쪽 혹은 오른쪽으로 마우스를 조작하는것과 화면주위로 마우스지시자를 빨리 이동하는 방법을 정의한다.

개성적인 설정

Personalization부분에는 건반이나 마우스의 동작특성, 언어와 방언, 전자우편기정설정, 암호화, 통과암호를 비롯하여 개별적인 요구에 따라 KDE 2.1.1 을 구성하는 여러가지 방법이 있다.

- Accessibility 는 번쩍임(visible flash)으로 체계종(bell)의 변경과 건반설정조절, 유표이동을 위한 건반을 설정한다.
- Country & Language 를 설정하여 국부적인 계산체계들과 화폐 및 시간에 대해서 KDE 가 리용하는 언어와 방언을 설정한다.

주의	미국식영어가 아닌 다른 언어를 설정하고 Red Hat Linux 7.1 을 설치할 때 영어를 설치하지 않았다면 이 책과 함께 제공된 Publisher's Edition 설치 CD-ROM 으로부터 알맞는 RPM 을 설치할수 있다. 실례로 우크라이나어를 설정하기전에 **rpm -i** 지령을 리용하여 **kde-i18n-Ukrainian-2.1.1-2.nosrch.rpm** 을 설치해야 한다.

- Crypto 를 설정하면 리용해야 할 암호화 및 복호화방안들을 설정할수 있다.
- Email 을 설정하면 ID 와 봉사기, 전자우편의뢰기정보를 설정할수 있다.
- Keyboard Layout 는 브라질어, 뽀르뚜갈어, 로씨야어와 같은 언어들을 비롯하여 X Windows 체계가 허용하고 제공하는 익숙된 많은 민족어건반을 리용할수 있게 한다.
- News Ticker 는 KDE 2.1 의 새로운 특성이다. 그것은 주기적으로 표제들을 조사하는 싸이트들을 전용화할수 있게 한다. News Ticker 로 작업할수 있는 현재 게재된 소식싸이트들은 http://www.webreference.com/services/news/ 에 배치된다.
- Passwords 는 현시장치에 통과암호가 어떻게 나타나는가를 설정한다. Remember Password 항목을 선택하면 KDE 는 정해 진 시간에 뿌리통과암호를 요구하는 KDE 편의프로그람의 특급사용자통과암호를 기록한다.

무릎형콤퓨터전원조종

무릎형콤퓨터에서 가동하는 Linux 는 축전지수명을 늘이기 위해서 전원조종을 요구한다. PowerControl 항목에는 전원소비를 관리하며 전원의 축전지공급준위를 측정하는 편의프로그람들이 있다. 특히 다음과 같은 보조항목을 포함한다.

- Battery Monitor 보조항목은 콤퓨터에 현재 설치된 축전지의 전원준위를 측정하는 조종판에 아이콘들을 추가하게 한다. 여기서는 고급한 전원관리데몬이 현재 설치되였다고 가정한다.
- Energy 보조항목은 Standby, Suspend 혹은 Off 방식으로 현시장치를 설정할수 있는 계수기(시계)들을 구성한다.
- Laptop Power Control 보조항목은 무릎형콤퓨터를 Standby 혹은 Suspend 상태로 만들수 있다. 실제적인 효력은 전문무릎형콤퓨터에 따른다.

주의	Laptop Power Control 을 설정하기전에 Advanced Power Management 에서 허가들을 수정해야 한다. 뿌리사용자로서 chmod u+s/usr/bin/apm 지령을 실행한다.

- Low Battery Critical 과 Low Battery Warning 은 축전지전력이 몇분 유지될수 있는가 하는 통보문을 제시한다. 리티움이온축전지들은 아주 정밀하여 다른 형태들의 콤퓨터축전지들보다 더 적은 전력을 방출한다.

음성조종

KDE 에는 음성을 구성하기 위한 여러가지 수법들도 있다. KDE Control Center Sound 항목은 4 가지 보조항목들을 가진다.

- Midi 는 콤퓨터우에서 음성을 만드는 여러가지 Sound Blaster 와 같은 기계장치구동기들을 구성할수 있게 한다.
- Mixer 는 검출된 음성기판들의 혼합을 증가시킨다. 음성기판들이 특별한 능력이 있으면 검사된 혼합기들과 혼합기당 장치수를 증가시킬수 있다.
- Sound Server 는 음성기판에 주어 진 자원들을 구동한다.실례로 완전 2 중(full duplex)조작은 음성이 수화기와 고성기로 동시에 나오게 한다.
- System Bell 설정은 체계종사건이 있을 때 울리는 종의 음량, 음도, 지속시간을 구성한다.

체계사건

System 항목은 5 개 보조항목들에 따르는 많은 기본파라메터들을 구성하거나 재현하게 한다.

- Boot Manager 는 콤퓨터에 설치된 각이한 체계들과 핵심부들을 설정할수 있게 Linux Loader(LILO)를 구성하고 도형대면부말단을 제공한다. 재기동할 때 LILO 는 자기가 요구하는 조작체계를 선택하고 구동할수 있게 한다.
- Date & Time 은 콤퓨터에 정의한 시간, 날자 혹은 세계시간을 수정한다.
- KDE System Control 은 콤퓨터하드웨어와 관련된 설정들을 재현하게 한다.
- Login Manager 는 KDE 가입화면을 변경시키기 위해 이미 본 설정상태로서 작업한다.

- Session Manager 는 탈퇴할 때 확인요구와 가입할 때 이전의 대화조종을 복귀하기 위한 기정항목들을 설정한다.

Web 열람

Web Browsing 항목은 cookies 와 proxies, 여러 형태의 열람기로 Web 통신들을 구성한다. 특히 다음과 같은 보조항목을 구성할수 있다.

- Cookies 는 열람기와 각이한 Web 싸이트사이에서 교환되는 정보를 구성한다. 령역으로 이것을 전용화할수 있다.
- Enhanced Browsing 은 각이한 Web 싸이트들에 대한 여러가지 열쇠단어들을 구성한다. 이것은 하나의 열람기로서 Konqueror 에만 적용한다.
- Konqueror 는 Java 리용을 비롯하여 Web 페지들의 현시를 조종한다.
- Netscape Plugins 는 여러 형태의 파일과 련관된 접속모듈을 제시한다.
- Proxies 는 proxy 봉사기콤퓨터를 통하여 인터네트나 기타 Web 망들에 결합된다.
- User Agent 는 Web 봉사기가 열람기를 조사하는 방법을 조종한다.
- Windows Shares 는 Microsoft Windows 망에서 국부콤퓨터를 Samba 의뢰기로 설치한다.

차림표와 탁상의 갱신

조종쎈터가 제공하는 구성항목들을 내놓고도 또한 차림표, 탁상아이콘 및 조종판을 변경시켜 KDE 환경을 갱신할수 있다.

이러한 항목을 갱신하기 위해서는 우선 KDE 에 추가하려는 항목들을 서술하는 KDE Link 파일을 만들어야 한다. 이 장의 《KDE Link 파일만들기》항목에 KDE Link 파일을 도형방식으로 만드는 방법을 주었다.

장치나 응용프로그람을 추가하는 KDE Link 파일이 있으면 자기가 만든 등록부에서 그 파일아이콘을 끌어다가 KDE 를 구성하는 위치에 놓을수 있다. 즉

- 조종판에 그 응용프로그람이나 장치의 아이콘들을 추가하기 위하여 조종판에 KDE Link 파일아이콘을 놓는다(조종판으로부터 그것을 재정리하거나 삭제하려면 임의의 조종판아이콘을 오른쪽찰칵한다.).
- KDE 를 기동할 때마다 자동적으로 그 응용프로그람을 실행시키기 위하여 탁상의 Autostart 폴더에 KDE Link 아이콘을 놓는다.
- 임의의 시각에 아이콘을 찰칵하여 그 장치나 응용프로그람에 즉시 호출할수 있도록 KDE 탁상에 KDE Link 파일을 놓는다.

일러두기 다른 폴더에 아이콘을 놓을 때 그 위치의 파일을 복사하고 옮기고 붙이는 작은 차림표를 주의해야 한다. 대체로 파일을 놓으려는 원래장소에 그 파일을 복사하려면 KDE Link 파일에서 Copy 를 선택하는것이 제일 좋다.

- 마지막으로 KDE 차림표를 정의하는 등록부가운데서 어느 한 등록부에 KDE Link 파일을 끌어다 놓을수 있다. KDE Link 파일이 정의한 응용프로그람은 KDE 차림표의 부분으로 될수 있다. 뿌리사용자로 가입하면 전체 사용자의 기본차림표를 변경하기 위해 /usr/share/applnk 등록부에 새로운 KDE Link 파

일을 배치할수 있다. 동시에 보조차림표에 그것들을 배치하기 위해 /udr/share/applnk 보조등록부들중 어느 한 보조등록부에 그 파일을 놓을수 있다.

KDE 편의프로그람의 리용

KDE 는 본문과 도형의 보기, 대상과제의 관리, Linux 체계의 관리에 리용할수 있는 몇개의 표준편의프로그람들을 포함하고 있다. 이 절에서는 많이 쓰이는 편의프로그람들을 소개한다.

본문과 도형처리의 보기

KDE 는 본문편집기, 파일현시기, 영상편집기(paintbrush 와 같은)를 가지고 있다. 기본차림표에서 Editors → Text Editor 를 선택하여 표준본문편집기 KEdit 를 호출할수 있다. 또한 기본차림표에서 Editors → Advanced Editor 를 선택하여 고급한 본문편집기 KWrite 도 호출할수 있다.

또한 Grapihics 보조차림표로부터 KView 도형현시기(aka Image Viewer), KDEPivmap2Bitmap, Xpaint 를 비롯하여 여러가지 영상현시기를 호출할수 있다. 또한 Graphics 보조차림표로부터 영상편집기인 The GIMP 를 호출할수도 있다.

Konqueror 에서 적당한 파일을 오른쪽찰칵하면 이 응용프로그람들을 리용하여 그 파일을 열수 있다. 실례로 찰칵하고 튀여나오기차림표에서 Open With 를 찰칵한다. 그러면 그림 11-18 에서 보는것처럼 본문파일을 열수 있게 구성한 모든 응용프로그람에서 임의의것을 선택할수 있다.

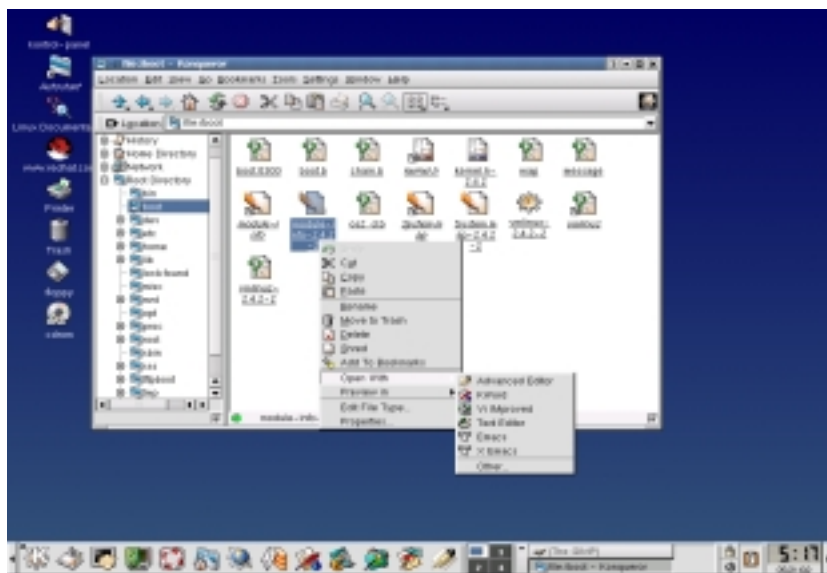



그림 11-18. 응용프로그람으로 본문파일을 열기 위해 Konqueror 를 리용

다른것을 선택하려면 Other 를 찰칵해야 한다. 그러면 Open With 대화칸을 열고 KDE 기본차림표에서 보여 주는 응용프로그람가운데서 어느 하나를 선택할수 있다.

시간관리

Linux 에는 Microsoft 의 Outlook 혹은 Project 와 같은 복사제품이 없지만 KDE 에는 시간을 관리하는 몇가지 도구들이 있다.

KDE 에는 한개의 사건력서와 우선권화면목록, 일정작성 등을 할수 있는 KOrganizer 라는 시간관리도구도 있다. 기본차림표에서 Applications → Organizer 를 선택하여 이 도구를 호출할수 있다.

그림 11-19 는 기본 KOrganizer 화면을 보여 준다. 이 화면에서 각이한 날자에로 이동하거나 작업목록을 추가하거나 각이한 속성에 따라 일정작성을 추가할수 있다.

Korganizer 외에도 KDE 에는 작업하는 여러가지 대상과제들에서 얼마나 많은 시간을 소비하는가를 추적하기 위해 시계를 시동 및 정지할수 있는 시간추적기(time tracker)도 있다. 이 도구는 Utilities → Personal Time Tracker 를 선택하여 리용할수 있다.

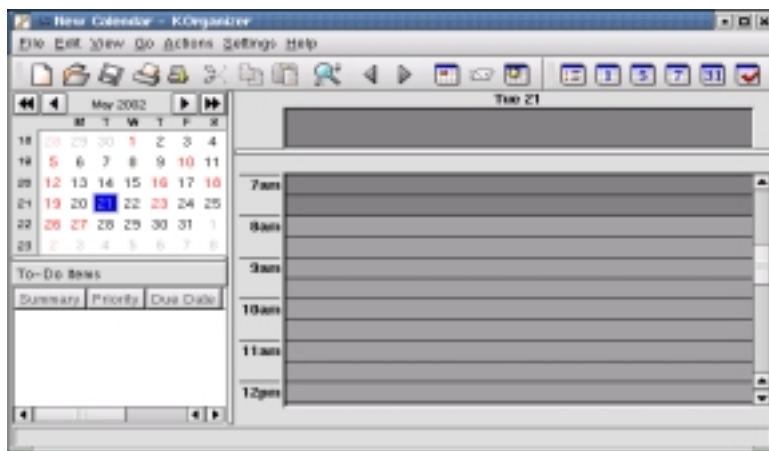



그림 11-19. 한개의 력서, 한개의 목록칸, 여러개의 순서짜기도구들이 있는 KOrganizer 프로그람

KDE 는 또한 간단한 Address Book 편의프로그람을 가지고 있다. 콤퓨터로 작업하면서 련계할수 있는 동무나 동업자들에 대한 함축된 정보 및 주의사항을 기록할 때 이 프로그람을 리용한다. Address Book 는 매 사람들에 대한 전화번호, 팍스번호, 전자우편주소를 기록한다. 이 프로그람을 리용하려면 Utilities → Address Book 을 선택하여야 한다.

KDE 사무용도구

KDE 2.0 은 Koffice 묶음의 사무도구들을 갖추고 있다. 이 도구들에는 Microsoft 의 Office, Sun Star Office 혹은 Applix AnywareOffice 와 같은 여러개의 종합된 응용프로그람들이 있다. Koffice 에는 KDE 기본차림표의 Office 를 선택하여 호출할수 있는 5 개의 기본도구들이 있다.

- KWord 는 Microsoft 의 Word 파일들을 수용할수 있는 틀제작기(FrameMaker) 형식의 문서처리기이다.
- KSpread 는 그림 11-20 에서 보여 준것처럼 반점식별형식(CSV)으로 자료파일

뿐아니라 Microsoft 의 Excel 파일들도 수용할수 있는 KOffice 작업판응용프로그람이다.

- KPresenter 는 Microsoft 의 PowerPoint 파일들도 수용할수 있는 KOffice 부속응용프로그람이다.

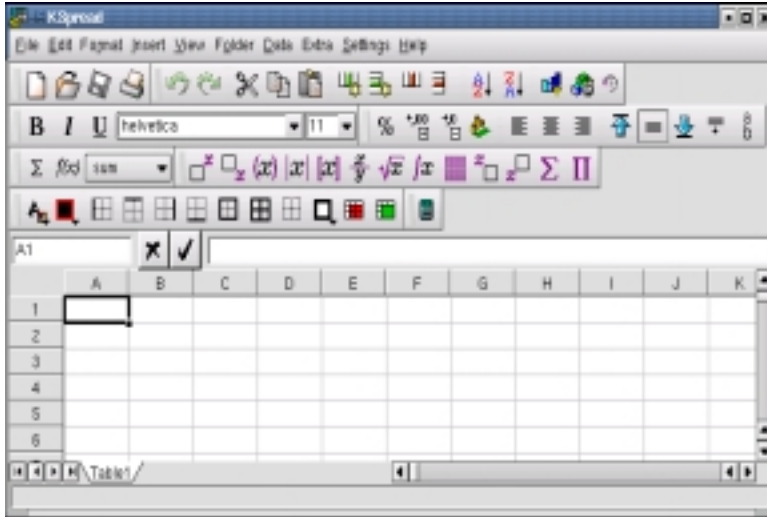



그림 11-20. KOffice 의 작업판응용프로그람인 KSpread

- KChart 는 도식(흐름도)들과 도형을 만들수 있는 도구이다.
- KIllustrator 는 콤퓨터지원설계에 적합한 벡토르-도형처리프로그람이다.

KOffice Workspace 에 이 프로그람들을 모두 묶어 놓을수 있는데 이것으로 한개의 프로그람창문에 있는 프로그람가운데서 임의의 프로그람을 기동할수 있다. 한 장소안에 KOffice 응용프로그람들이 모두 있으면 응용프로그람들사이의 자료를 더 쉽게 공유할수 있다.

기타 유용한 도구들로서는 Krayon, 영상처리프로그람인 Kugar, 보고서작성프로그람인 Katebase, 자료기지관련프로그람인 형식관리를 위한 Kformula, 흐름도를 위한 Kivio 등이 있다. Krayon 과 Kugar 는 2001 년에 KOffice 의 새 판본들로 개정되였다.

Linux 체계의 관리

KDE 는 새로운 체계관리도구들도 추가한다. 여기에서 중요한것은 RPM 프로그람묶음, 체계의 사용자들과 그룹, 체계에서 실행되고 있는 프로세스, Linux 의 System V 초기화스크립트를 관리하는것이다.

이 편의프로그람들의 대부분은 기본차림표의 Utilitles 와 System 항목에서 리용할수 있다.

Linux 관리과제들에서 본문형식의 구성파일로 작업할 필요가 제기되지만 Kpackage 가 있으면 과제들을 훨씬 더 쉽게 완성할수 있다. KPackage 를 기동하려면 System → Package Manager 를 선택하여야 한다. 그러면 기본 KPackage 창문이 나타나므로 체

계에 있는 모든 RPM 프로그람묶음정보를 읽을수 있다(그림 11-21).

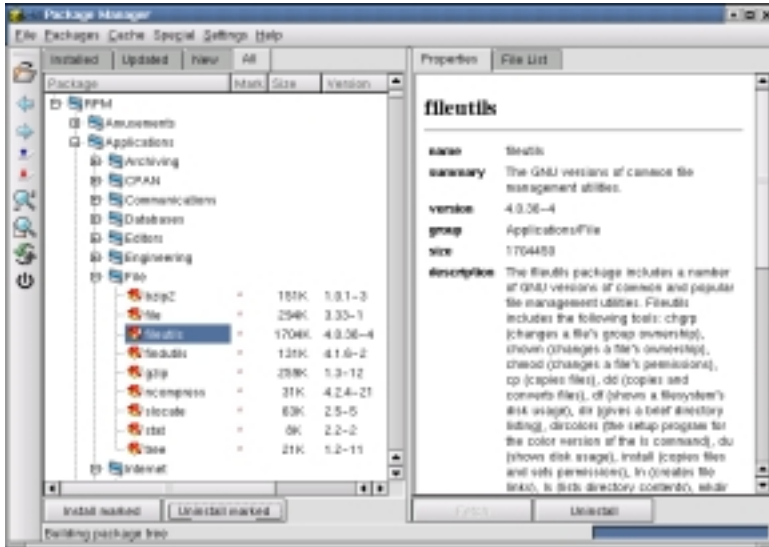



그림 11-21. 기본 KPackage 창문에 설치된 모든
RPM 프로그람묶음들의 부류별목록화

이 창문에 설치된 프로그람묶음을 재현시킬수도 있고 체계관리를 위한 프로그람묶음들과 파일을 탐색하거나 혹은 새로운 RPM 프로그람묶음들을 설치할수도 있다.

▶미리 알아두기

이제는 각이한 도형 Linux 탁상들을 잘 알게 되였으므로 X Windows 환경의 보다 복잡한 구성에로 넘어 갈수 있다.

제 12 장《고급한 X Windows 의 구성》에서는 X Windows 의 개별적요구들과 취미들을 만족시키는 창문이나 탁상관리도구를 탐색하는 방법을 본다. 이 과제는 X Windows 기동순서는 물론 특수상태에서 XFree86 봉사기를 구성하는 방법도 포괄하고 있다.

제 12 장. 고급한 X Windows 의 구성

이 책에서 이미 본바와 같이 X Windows 환경은 다른 GUI 조작체계들에서 일반적으로 볼수 없는 유연성과 객관성을 제공한다. 물론 이러한 유연성은 일정한 복잡성을 초래하지만 X Windows 의 유연성을 완전히 리용하려면 때때로 복잡한 구성방법을 리용하여야 한다.

이 장에서는 X Windows 의 기본구성요소들을 상세히 설명하고 그것들을 어떻게 전용화할수 있는가를 설명한다. 이것은 X 봉사기가 어떻게 동작하는가를 보여 주는 XFree86 판본 4 의 XF86Config 파일로 기동할수 있다. 이 파일에서는 리용중에 있는 비데오기판과 현시장치의 형(해상도와 색계조를 포함)으로부터 리용중에 있는 입구장치들(마우스와 건반)에 이르기까지의 모든것을 정의할수 있다.

X 봉사기가 필요한 구성을 완전히 끝내면 X Windows 가 어떻게 기동하는가 즉 처음에 어떤 프로그람들이 기동하며 어떤 동작이 취해 지는가를 조종할수 있으며 창문관리기가 어떻게 시작되는가도 조종할수 있다. 또한 X 자원자료기지는 창문에서 쓰이는 색이나 서체, 동작에 따르는 창문들의 변화와 같은 설정들을 조종할수 있는 기구를 제공한다. 이러한 설정(그리고 기타설정)은 전체 혹은 매 응용프로그람으로 정의할수 있다.

XF86Config 파일

XFree86 의 구성은 XF86Config 파일에서 진행한다. 이 파일은 보통 자기의 배포물에 따라 /etc/XF86Config 나 /etc/X11/XF86Config 에서 찾을수 있다. Red Hat Linux 7.1 에서는 /etc/X11 에 그 파일이 존재한다. 선택적으로 /usr/X11R6/lib/X11 /XF86Config.eg 파일과 같은 일반적인 구성파일로 기동할수 있다.

이 파일이 어디에 있는지 모르는 경우 그것을 찾으려면 다음의 지령을 입력해야 한다.

```
# locate XF86Config
```

XF86Config 파일에는 건반, 마우스, 현시장치정보를 비롯하여 X 봉사기의 조작과 관련된 정보들이 있다. 이 파일은 Xconfigurator 나 xf86config 와 같은 XFree86 구성 응용프로그람들에 의하여 발생된다.

그러나 때때로 X Windows 환경을 어느정도 조절하려면 본문편집기로 XF86Config 파일을 직접 편집할수 있다.

주의 이 책에서 보는바와 같이 XFree86 판본 4 봉사기는 XFree86 판본 3 에서처럼 많은 도형처리기판을 위한 구동프로그람들을 가지고 있지 않다. 그러므로 거의 모든 구성들에는 XF86Config 와 XF86Config-4 파일이 다 들어 있다. /etc/X11 등록부에 이 파일이 다 있는 경우 다음의 지령들을 /etc/X11/XF86Config-4 파일에서 리용할수 있다.

XF86Config 파일은 여러개의 부분으로 갈라 진다. 일부 부분에서는 작업하는 GUI 가 필요 없다. 그리고 XFree86 판본 4 에서는 이 부분들의 순서가 그리 중요하지 않다.

Files 이 항목은 XFree86 이 서체, 적재할수 있는 모듈, 색인과 같은 보조적인 파일들을 찾을수 있는 장소를 정의한다.

ServerFlags 이 항목은 건반렬을 조종하는 방법과 같이 X 봉사기의 특성들을 유효로 만들수 있거나 또는 무효로 할수 있는데 리용된다.

Module 이 항목은 X 봉사기의 기동과정에 적재되는 모듈들을 정의한다. 이것들은 서체주사모듈들을 포함한다. 대부분의 경우에 X 봉사기로 작업하며 표준서체만으로는 이 작업이 충분하기때문에 XF86Config 파일에는 Module 항목이 필요 없다.

InputDevice 이 항목은 입력장치들을 정의하는데 건반과 마우스를 포함한다. Keyboard 와 Pointer 항목이 없으면 이것은 필요 없다.

Keyboard 이 항목은 건반규약을 지정하는데 그것은 반복속도와 같이 건반작성을 정의하는 방법과 기타 건반특성들을 조종한다. 건반에 대한 InputDevice 항목이 없으면 이것은 필요 없다. 이것은 XFree86 판본 4 로 작업할수 있다.

Pointer 이 항목은 리용되고 있는 마우스의 형태, 마우스가 들어 가는 포구형태, 마우스에 대한 단추동작방식을 정의한다. 건반에 InputDevice 항목이 없으면 이것은 필

요 없다. XFree86 판본 4 로 작업할수 있다.

VideoAdaptor 이 항목은 작업과정에 일부 큰 체계들이 요소로 되는 비데오정합기 즉《primitive》으로 비데오를 지원한다. 이 책을 쓸 당시는 리용되지 않았다.

Monitor 이 항목은 현시 장치의 특성들을 정의한다. 정보들을 Modes 항목에 포함시킬수 있다.

Device 이 항목은 쓸수 있는 도형처리장치들(현시 장치정합기)을 정의한다.

Modes 이 항목은 현시장치의 재생속도와 분해능을 정의한다. Monitor 항목에 포함시킬수 있다.

Screen 이 항목은 비데오정합기들에 현시장치들을 련결한다. 또한 일반 SVGA 봉사기, 전용봉사기 및 흑색봉사기와 같은 X 봉사기들의 동작을 정의한다. 분해능과 색계조들을 바꿀수 있는 다중현시장치의 보조항목들을 포함시킬수 있다.

ServerLayout 이 항목은 입력장치 즉 건반과 마우스를 정의하는 화면들을 련결한다.

DRI 이 항목은 도형처리전용워크스테이션들에 대해서 가장 일반적인 Direct Rendering Interface(DRI)에 필요한 임의의 어떤 특수한 설정들을 처리한다. DRI 구성은 이 책에 들어 있다. 더 상세한 정보에 대해서는 Web 싸이트 http://www.xfree86.org/current/DRI.html에서 볼수 있다.

Vendor 이 항목은 판매자정의 설정기능들에 리용된다. 이 책을 쓸 당시에는 리용되지 않았다.

Files

Files 항목은 중요파일들이 체계우의 어디에 적재되는가를 정의하는데 리용된다. 다음에 보여 주는것이 설명문이 없는 Files 항목의 표본이다.

주의 XF86Config 파일에서는 설명문들이 하쉬표식(#)으로 시작된다. XFree86 은 설명문에 들어 있는 내용들을 모두 무시한다.

```
Section  "Files"

    RgbPath    " /usr/X11R6/1ib/XII/rgb"

    FontPath   " /usr/XIIR6/lib/XII/fonts/1ocal/"
    FontPath   " /usr/X11R6/1ib/XII/fonts/misc/"
    FontPath   " /usr/X11R6/1ib/XII/fonts/75dpi/:unscaled"
    FontPath   " /usr/X11R6/1ib/XII/fonts/l00dpi/:unscaled"
    FontPath   "/usr/XI1R6/1ib/XII/fonts/Typel/"
    FontPath   " /usr/X11R6/1ib/XII/fonts/Speedo/"
    FontPath   " /usr/X11R6/1ib/XII/fonts/75dpi/"
    FontPath   " /usr/X11R6/1ib/XII/fonts/l00dpi/"

#   ModulePath       "/usr/X11R6/lib/modules"

EndSection
```

다른 6개의 항목들에 적용할 때 주의해야 할 여러가지 기본규칙들이 있다.

- 매 항목은 section 행으로 시작된다.
- 매 항목은 EndSection 행으로 끝난다.
- Section 행에서 항목이름이 인용괄호안에 정의된다(이 경우에는 Section "Files").

이 항목에는 일반적으로 쓰이는 3개의 지령들이 있는데 이것들을 표 12-1에 주었다.

표 12-1. **Files 지령**

지 령	효 과
RgbPath	이 지령은 **.txt** 혹은 **.db** 확장자가 아닌 RGB(적-청-록)자료기지의 이름을 정의한다. RGB 파일은 다른 색갈들과 관련된 적, 청 및 록색의 크기를 정의한다. 보통 기정값은 RGB 자료기지의 이름을 옮기거나 변경시키지 않는 한 유지될수 있다(이것은 그리 좋은 생각은 아니다. 여러가지 응용프로그람은 자료기지를 만들고 표준위치에서 그것을 찾을수 있다고 볼수 있다.)
FontPath	이 지령은 X Windows 가 체계에 설치된 X 서체들을 어데서 찾을수 있는가를 정의할 필요가 있을 때마다 리용한다(척도화될수 없는 bitmap 서체들은 **:unscaled** 을 등록부의 마감에 첨부하였다.). 체계우에 새로운 서체등록부들을 추가하면 여기에서 그것들에 대한 **FontPath** 지령을 추가해야 한다.
ModulePath	이 지령은 동적으로 적재되는 모듈들을 지원하는 Linux 와 같은 조작체계에서 리용된다. 이 지령은 모듈들을 찾는 장소를 가리킨다. 대부분의 XFree86 판본들에서는 이 지령이 나타나지 않거나 실례에서와 같이 설명된다. 모듈에 대한 기정경로는 **/usr/X11R6/lib/modules** 이며 이 값을 변경해야 할 때에만 이 지령을 주어야 한다.

기타 서체모듈들은 /usr/X11R6/lib/modules/fonts 등록부에서 리용할수 있다.

ServerFlags

ServerFlags 항목은 X 봉사기의 특성들을 리용할수 있게도 하고 없게도 한다. Server Flags 항목의 실례들을 설명이 없이 다음과 같이 표시한다.

```
Section "ServerFlags"

# Option      "NoTrapSignals"

# Option      "DontZap"
# Option      "DontZoom"

# Option      "DisableVidModeExtension"

# Option      "AllowNonLocalXvidtune"

# Option      "DisableModInDev"
```

```
# Option       "AllowNonLocalModInDev"

# Option       "AllowMouseOpenFail"

# Option       "blank time"   "10"
# Option       "standby time"        "20"
# Option       "suspend time"        "30"
# Option       "off time"     "60"

# Option       "EstimateSizesAggresively" "0"

# Option       "NoPM" "false"
# Option       "Xinerama" "true"

EndSection
```

이 지령들의 임의의것을 능동화하려면 설명문을 없애야 한다. 비능동화하려면(보통 일반적인 상태) 설명문이 있어야 한다. 정의된 선택항목들은 여기에서 ServerLayout 단락의 선택항목들로 바뀐다.

이 실례에서 보게 되는 지령들은 표 12-2에 준 효과를 가진다.

표 12-2. ServerFlags 지령

지 령	효 과
NoTrapSignals	이 지령은 수정작업에 쓰인다. 오유가 접수되였다는 신호가 나타나면 봉사기는 디스크에 있는 그 파일내용을 기억기에 배치하고 오유가 나타난 기본구역을 화면에 제시한다.
DontZap	보통 Ctrl+Alt+Backspace 의 건 조합은 X Windows 를 취소하고 조종탁에 돌아 가도록 한다. 이 지령을 능동으로 하면 Ctrl+Alt+Backspace 는 무시되고 현재처리중인 응용프로그람에로 넘어 간다.
DontZoom	보통 Ctrl+Alt+Keypad Plus 의 건 조합은 분해능을 증가시킬 때마다 가능한 화면분해능을 통하여 순환된다. Ctrl+Alt+Keypad Minus 는 분해능을 감소시키는데 리용된다. 이 건 조합들이 응용프로그람에 따라 다른 목적에 요구되면 X Windows 는 이 건 조합을 무시하고 그것들을 응용프로그람에 넘기도록 이 행을 설명하지 않는다.
DisableVidModeExtension	이 지령은 xvidtune 의뢰기가 비데오현시장치를 켜지 못하게 한다(Xvidtune 는 이 책에서 론의하지 않는다. 이 프로그람에 대한 문서는 xterm 창문을 열고 **man xvidtune** 지령을 입력하여 찾아 볼수 있다.). 이 항목을 설명하지 않고 그것을 기정으로 남겨 둘수 있다.
AllowNonLocalXvidtune	앞의 지령과 관련시켜 이 지령은 비국부적인(다른말로 망의 밖에서) xvidtune 의뢰자들이 X Windows 체계를

표계속

지　　령	효　　과
	호출하도록 한다. 보안상 문제로 하여 이 지령을 비능동상태로 해주는것이 좋다.
DisableModInDev	이 지령을 능동상태로 하면 X Windows 체계가 실행할 때 건반과 마우스설정들을 동적으로 변경시키는것이 불가능하다.
AllowNonLocalModInDev	이 지령을 설명하지 않으면 비국부적콤퓨터들이 마우스와 건반설정들을 바꿀수 있다.
AllowMouseOpenFail	마우스로 작업하고 있지 않는 경우 X 봉사기가 기동하도록 한다.
"blank time""10"	정의된 종결시간이 되면 화면보호기가 능동화된다(이 경우에는 종결시간이 10분이다).
"standby time" "20"	정의된 종결시간이 되면 현시장치가 정지(standby)방식으로 넘어 간다(20분).
"suspend time" "30"	정의된 종결시간이 되면 현시장치가 중지(suspend)방식으로 넘어 간다(30분).
"off time" "60"	정의된 종결시간이 되면 현시장치가 끄기(off)방식으로 넘어 간다(60).
EstimateSizesAggresively	두번째 비데오정합기를 검출하는 BIOS 문제들이 있으면 이 변수를 2로 설정하도록 방조할수 있다.
NoPM "false"	전원관리를(Power Management)를 무시한다.
Xinerama "true"	개개의 도형처리기판을 갖춘 다중현시장치들을 알려 준다.

Module

Module 선택항목은 봉사기와 서체확장의 기본파라메터들을 서술한다. 모듈들은 작업과정에 도형처리사용자대면부를 요구하지 않는다. 서문화가 없는 Module 항목에 대한 실례를 아래에 주었다.

```
Section "Module"
# Load "dbe"
# SubSection "extmod"
# Option "omit XFree86-DGA"
# EndSubSection
# Load "type1"
# Load "freetype"
EndSection
```

능동화하려는 이 지령가운데서 어느것도 서문화하지 않았다. 즉 그것들을 능동화하지 않으면 서문화되지 않는다(보통 지정상태). 이 실례들에 있는 지령들은 표 12-3 에서 서술한것처럼 여러가지 효과를 가진다.

표 12-3. Module 지령

지 령	효 과
Load "dbe"	련속적인 영상들이 여러 완충기들에 적재되도록 하는 Double Buffer Extensions 를 적재한다.
Option"omit XFree86-DGA"	Direct Graphics Access 확장을 무시한다.
Load "type1"	PostScript type 1 서체모듈을 적재한다.
Load "freetype"	명명된 True Type 클론을 적재한다.

기타 확장모듈들은 /usr/X11R6/lib/modules/extensions 등록부에서 리용할수 있다.

InputDevice-Keyboard

InputDevice 항목들은 마우스(지시자)와 건반을 정의한다. InputDevice 가 XFree86 판본 4 에 대해 기정이면 Keyboard 와 Pointer 모듈들은 계속 리용할수 있다. 보통 두개의 InputDevice 항목들이 있다. 하나는 건반과 지시장치에 대한것이다.

예상했던것처럼 InputDevice 항목은 건반형태와 규약을 비롯하여 건반기능에 관한 정보를 정의한다. 이 항목이 기정으로 더이상 쓰이지는 않지만 XFree86 판본 4 는 여전히 인식한다. 비록 Red Hat Linux 7.1 이 XFree86 판본 4 를 리용한다고 해도 Keyboard 와 Pointer 항목들은 Red Hat Linux 7.1 의 XF86Config 구성파일부분에 속한다. InputDevice 항목의 실례를 아래에 보여 주었다.

```
Section "InputDevice"
# Option      "Protocol"    "Xqueue"

# Identifier  "Keyboard1"
# Driver       "keyboard"

# Option     "AutoRepeat"   "500 5"

# Option     "XkbDisable"

# Option     "Xleds"        "1 2 3"

# Option      "LeftAlt"     "Meta"
# Option      "RightAlt"    "ModeShift"
# Option      "RightCtl     "Control"
# Option      "ScrollLock"  "Compose"
# Option      "XkbDisable"
# Option      "XkbModel"    "pc101"
```

```
# Option       "XkbModel      "pc102"
# Option       "XkbModel"      "pc104"
# Option       "XkbModel"      "pc105"
# Option       "XkbModel"      "pc106"
# Option       "XkbModel"      "microsoft"

# Option       "XkbLayout"     "us"
# Option       "XkbLayout"     "de"
# Option       "XkbVariant"    "nodeadkeys"
# Option       "XkbOptions"    "Ctrl:swapcaps"

# Option       "XkbRules"      "xfree86"
# Option       "XkbKeymap"     "xfree86(us)"

EndSection
```

이것은 복잡한 항목처럼 보이지만 표 12-4 에서 보는것처럼 아주 단순하다.

표 12-4. Keyboard 지령

지 령	효 과
"Protocol"" Xqueue"	Xqueue 규약은 여러 Sun Solaris 봉사기들에서 리용된다.이 행이 나타난면 XFree86 은 거의 모든 다른 체계에서 작업할수 있게 Standard 로 기정화한다.
Identifier, Driver	관리기가 건반과 건반구동프로그람에 이름을 할당한다.
AutoRepeat	이 지령은 자동반복을 시작하기전에 건반눌리움시간을 얼마만큼 설정하며 건반이 얼마나 반복되여야하는가를 정의하는데 리용된다. 값들은 모두 미리초이다. 앞실레에서 AutoRepeat 500 5 는 자동반복은 건반이 눌리운지 500ms 후에 시작되여야 하며 그다음에는 5ms 간격으로 반복되여야 한다는것을 나타낸다
Xleds	이 지령은 xset 지령을 리용하는 사용자가 조종할수 있는 NumLock, Caps Lock 와 같은 건반 LED 들을 정의한다.
LeftAlt,RightAlt, RightCtl, ScrollLock	X Windows 는 보통 표준 PC 건반과 명백하게 다른 건반들을 가진 Unix 망작업기에서 기동한다. 이 건반들은 Meta, ModeShift, Compose 및 ModeLock 와 같은 전용건들을 가지고 있다. Unix 와 기타 응용프로그람이 이 건들을 요구한다면 이 지령들을 서문화하지 말아야 한다.
XkbDisable	이 행이 서문화되여 있지 않다면 XKB 로 불리우는 X Windows 의 확장자는 무시된다. XKB 가 허용되면 XKB 는 일련의 지령들을 리용하여 일반적인

	X 봉사기들에 건반배치를 정의한다.
XkbModel	이 지령은 XKB 가 건반모델을 정의할수 있을 때 리용한다. 표준 U.S 건반은 pc101 이다. U.S "Windows"건반은 pc104 이다. Micresoft Natural 건반은 microsoft 이다. 유럽건반들은 pc102 혹은 pc105 이다. 표준일본어건반은 pc106 이다.
XkbLayout	이 지령은 건반배정을 정의하도록 XKB 가 허용될 때 리용한다.
XkbOptions	이 지령은 Caps Lock 와 Ctrl 건반위치를 바꾸도록 XKB 가 허용될 때 리용될수 있다. 그러자면 이 지령의 값을 ctrl:swapcaps 로 설정하여야 한다.
XkbRules	건반배정에 대한 기본규칙을 설정한다. XFree86 봉사기를 쓸것을 요구한다.
XkbKeymap	이 지령은 XKB 가 건반작성정의를 적재하는것이 유효할 때 리용할수 있다.

일러두기 영문판이 아닌 건반을 설정하려면 http://www.linuxdoc.org 에 있는 HOWTO 문서에서 방조 받을수 있다. 특수언어건반을 구성하도록 방조할수 있는 여러개의 언어정의 HOWTO 가 있다.

InputDevice-Pointer

두번째 InputDevice 항목에서는 마우스관련구성을 취급한다. 앞서 이야기한것처럼 XFree86 판본 4 에 있는 마우스관련구성을 위해 InputDevice 나 Pointer 항목을 설정할 수 있다. Red Hat Linux 7.1 의 기정인 XF86Config 파일은 마우스를 구성하는 Pointer 항목을 가진다. 이 항목에서 마우스의 형태, 마우스가 련결된 포구, 마우스단추들과 관련된 행동규범과 같은 정보들을 설정한다.

```
Section "InputDevice"

# Identifier "Mousel"
# Driver    "mouse"

# Option "Protocol"     "PS/2"
# Option "Protocol"     "Xqueue"
# Option "Device"                "/dev/mouse"
# Option "BaudRate"     "9600"
# Option "SampleRate"   "150"

# Option "Emu1ate3Buttons"
# Option "Emulate3Timeout"     "50"

# Option "ChordMiddle"

EndSection
```

표 12-5 에 이 지령들의 내용을 주었다.

표 12-5. Pointer 지령

지 령	효 과
Identifier, Driver	관리기가 마우스와 장치구동기에 이름을 할당한다.
Protocol	이 지령은 마우스가 리용하는 규약을 정의한다. 전형적인 항목들에는 **Microsoft, Logitech, MouseSystems, BusMouse, PS/2** 및 **Auto** 가 있다(Linux 가 인식할수 있는 PnP 마우스에 대하여).
Device	이 지령은 마우스가 련결된 장치포구를 정의하는데 리용된다. 배포물을 설치하고 마우스를 정확히 구성하였다면 장치 **/dev/mouse** 는 마우스와 련결되며 이것을 리용할수 있다. 다른 한편 PS/2 마우스포구로는 **/dev/psaux** 를, Solaris 체계로는 **Xqueue** 를, DOS 와 Windows 의 **COM1** 로는 **/dev/ttyS0** 을, **COM2** 로는 **/dev/ttyS1** 을 지정한다. 기타 항목들은 추적볼(trackball)과 접촉화면(touchscreen)과 같은 장치들을 지정하는데 리용된다.
BaudRate	이 지령은 Logitech 상표가 붙은 마우스들에만 해당된다. 마우스문서를 조사해 보아야 한다.
SampleRate	이 지령은 Logitech 상표가 붙은 마우스들에만 해당된다. 마우스문서를 조사해 보시오.
Emulate3Buttons	이 지령은 단추가 두개인 마우스에만 해당된다. 서문화되지 않았을 때 왼쪽, 오른쪽단추를 동시에 누르면 세단추마우스의 가운데단추를 누른것과 같다. X Windows 는 세단추마우스를 리용하므로 두단추마우스는 세단추마우스의 역할을 충분히 할수 있다.
Emulate3TimeOut	이 지령은 두단추마우스의 두 단추를 동시에 찰칵하는것처럼 볼수 있게 두 단추들의 눌리움간격을 설정하는데 리용한다. 이 값은 **Emulate3Buttons** 가 유효할 때만 가능하다. 기정으로는 50 미리초이다. 마우스단추들을 동시에 찰칵하는데 이 시간간격이 너무 짧으면 이 시간간격의 값을 더 크게 정할수 있다. 이 지령은 미리초단위로 간주하므로 반드시 수값만을 주어야 한다. 즉 **Emulate3TimeOut 100** 이여야지 **Emulate3TimeOut 100ms** 라고 하면 안된다.
ChordMiddle	이 지령은 세단추 Logitech 마우스의 중간단추를 리용할수 있도록 한다. Logitech 마우스를 가지고 있지만 중간단추가 작업하지 않는다면 이 지령을 쓸수 있다.

USB 와 scroll 혹은 wheel 마우스들을 포함하여 다른 형태의 마우스들을 구성하는 정보들에 대해서는 제 7 장 《X Windows 의 설치와 구성》을 보아야 한다.

Monitor

지금까지 본 구성지령들은 대체로 자체프로그람들이고 리용하기가 쉽다. 자기의 콤퓨터와 접촉하는 매 영상장치들에 대해서는 개별적인 Monitor 항목이 필요하다. 다음과 같은 간단한 Monitor 항목을 고찰하자.

```
Section "Monitor"

Identifier  "monitor"
VendorName  "LG"
ModelName   "StudioWorks"

HorizSync   30-70

VertRefresh 50-160

End Section
```

표 12-6 에서는 Monitor 항목에 대한 각이한 지령들을 설명하였다.

표 12-6. Monitor 항목에 대한 각이한 지령

지 령	효 과
Identifier	이 지령은 Monitor 에 대한 특성들을 사용자가 선정한 이름으로 지정한다. 이것은 현시장치에서 언급한 **XF86Config** 파일의 다른 부분에서 리용된다.
VendorName	이 지령은 정의되고 있는 현시장치의 판매자를 확인한다. 이 값은 XFree86 의 조작에 영향을 주지 않으며 따라서 후날 그 정의를 확인해 볼수 있는 중요한 값을 지정하는데 가장 쓸모 있다.
ModelName	이 지령은 정의하고 있는 현시장치의 모델을 확인한다. 이 값은 XFree86 의 조작에 영향을 주지 않으며 따라서 후날 그 정의를 확인해 볼수 있는 중요한 값을 지정하는데 가장 쓸모 있다.
HorizSync	이 지령은 현시장치에 대해 kHz 의 수직동기범위를 정의한다. 실례에서 보여 준 범위를 정의할수도 있고 반점으로 구별한 불련속적인 값범위(실례로 15~25, 30~50)들의 목록을 제공할수도 있다. 현시장치문서를 참고하고 자기의 현시장치에 대한 정확한 값을 주는것이 중요하다. 정확하지 않은 값을 입력하면 현시장치가 파괴되거나 위험할수 있다.
VertRefresh	이 지령은 현시장치가 제공하는 Hz 단위의 수직동기재생속도를 정의한다. 실례에서 보여 준 범위를 정의할수도 있고 반점으로 구별한 불련속적인 값범위(실례로 40~50, 80~100)들의 목록을 제공할수도 있다. 현시장치문서를 참고하고 자기의 현시장치에 대한 정확한 값을 주는것이 중요하다. 정확하지 않은 값을 입력하면 현시장치가 파괴되거나 위험할수 있다.

Modes

콤퓨터에 접속한 모든 현시장치에 개별적인 Mode 항목들이 요구된다. 다음의 간단한 Mode 항목을 보자.

```
Section "Modes"

Identifier "Modes[0]"
Modeline "640x480" 46.02 640 656 760 832 480 490 498 522
Modeline "800x600" 71.91 800 808 928 1000 600 612 622 632
Modeline "1024x768" 117.53 1024 1088 1208 1360 768 783 796 829
Modeline "1600x1200" 200 1600 1616 1968 2080 1200 1200 1212 1253

EndSection
```

이러한 Modeline 들은 현시장치에 대해 가능한 방식들을 정의한다. 방식들은 해상도를 재생주기와 점(dot)박자, 시간선도들과 결합하여 현시장치에 어떻에 현시하겠는가를 결정한다. X 봉사기는 그것들을 적재하려고 할 때 XF86Config 파일에서 맞지 않는 방식들을 찾아서 삭제한다. 그러나 이 행들에 대한 정확한 정보를 얻기는 힘들다. 일반적으로 XFree86 구성쏘프트웨어가 이러한 동작을 만들도록 하는것이 가장 유익하며 그것들을 바꾸는것은 좋지 않다.

일러두기 다중현시장치가 있으면 복잡성을 피하기 위하여 대응하는 Monitor 항목들에 Modes 항목들을 삽입하는것이 더 좋다.

표 12-7 에서는 첫번째 Modeline 에 왼쪽으로부터 오른쪽으로 배치된 매 수들의 기능을 설명하였다.

표 12-7. Modeline 항목

항 목	설 명
640X480	화소단위의 화면해상도
46.02	mHz 단위의 화소박자
640	현시된 수평화소수
656	수평동기시작
760	수평동기끝
832	수평동기전체
480	표시되는 수직화소수
490	수직동기시작
498	수직동기끝
522	수직동기전체

첫번째 Modeline 행인

```
Modeline "640x640" 46.02 640 656 760 832 480 490 498 522
```

를 표시하는 또 다른 방법은 다음과 같다.

```
Mode "640△480"
    DotClock   46.02
    Htimings   640 656 760 832
    VTimings   480 490 498 522
EndMode
```

Device

XF86Config 파일은 다중 Device 항목들을 가질수 있는데 이것들은 다중도형처리정합기들이나 비데오기판들을 제공한다. Device 항목들은 X 봉사기가 제공할수 있는 비데오기판들을 서술한다. 일반적으로 봉사기는 이 정보들의 대부분을 장악하고 있지만 XF86Config 파일에서 결과들을 검색해 보아야 한다. 이 정보에 비데오하드웨어에 대한 고급한 기술적특성들이 있기때문에 이 정보들을 변경시킬 필요가 없다면 XFree86 구성 쏘프트웨어는 이 항목을 조종해야 한다.

Device 항목은 Screen 항목으로만 동작한다. Device 항목의 두가지 실례를 보자.

```
Section "Device"
     Identifier        "Generic VGA"

     VendorName  "Unknown"
     BoardName   "Unknown"
     Chipset          "generic"

#    VideoRam   256

#    Clocks     25.2 28.3
#    BusID     "PCI:1:0:0"
EndSection

Section "Device"

     Identifier  "MGA Millennium I"
     Driver     "mga"
     Option     "hw cursor" "off"
     BusID      "PCI:0:10:0"

EndSection
```

표 12-8에서는 Device 항목에 쓰이는 기본지령들을 설명하였다.

표 12-8. Device 지령

지 령	효 과
Identifier	Monitor 항목에서처럼 이 지령은 비데오기판을 인식하도록 XF86Config 파일에서 리용하는 이름을 제공한다.

표계속

지 령	효 과
VendorName	이 지령은 X 봉사기의 조작에 영향을 주지 않지만 하드웨어를 인식할수 있게 한다.
BoardName	이 지령도 역시 조작에 영향을 주지 않지만 장치정의를 인식하는데 쓸모가 있다.
Chipset	이 지령은 비데오소편을 확인한다. XFree86 이 하드웨어를 지원하지 않으면 generic 소편이 리용된다.
VideoRam	이 지령은 키로바이트단위로 비데오기억기용량을 정의한다. 이 정보를 제공하지 않으면 봉사기는 직접 비데오기판으로부터 그것을 확인한다.
Clocks	이 지령은 비데오하드웨어를 리용하는 박자소편을 정의한다.
ClockChip	이 지령은 비데오하드웨어에 의하여 리용되는 시계소편을 정의한다. 이것을 지정하면 시계소편이 모든 정보를 제공하기때문에 Clocks 행을 지적할 필요가 없다.
Driver	이 지령은 특수한 장치에 대한 구동프로그람의 이름을 정의한다.
Option	일부 구동프로그람들은 좀 더 복잡하게 구성될수도 있다. 실례에서 보여 준 "hw cursor" "off" 는 X Windows 에서의 유표배치와 관련된 일부 도형처리기판들의 특수한 과제들을 처리한다.
BusID	PCI 혹은 AGP 도형처리기판의 배치를 정의한다. 리용중인 한개이상의 도형처리기판이 있을 때 의미가 있다.

Screen

Screen 항목은 Monitor 와 Device 항목들에 있는 정보들을 모두 제시하는데 리용된다. 다중 Screen 항목들을 가질수 있다.

한개 Screen 항목의 실례는 다음과 같다.

```
Section "Screen"
    Identifier "Screen L"
    Device     "MGA Millenium I"
    Monitor    "monitor"
    Subsection "Display"
        Depth    24
        Modes    "1024x768"
        ViewPort 0 0
    EndSubsection
    Subsection "Display"
        Depth    32
        Modes    "800x600"
        ViewPort 0 0
    EndSubsection
EndSection
```

Screen 항목은 장치와 현시장치를 갖춘 X 봉사기에 련결하고 호출할수 있는 현시방식(분해능, 색계조 등)을 정의한다. 표 12-9 에서는 Screen 항목에 쓰이는 기본지령들을 설명하였다.

표 12-9. Screen 지령

지 령	효 과
Identifier	이 지령은 주어 진 **Screen** 에 대한 유일한 이름을 정의한다. **ServerLayout** 항목이 필요하면 이 **Identifier** 를 리용해야 한다.
Device	이 지령은 **Device** 식별자의 이름을 정의한다. 이것이 Device 항목에 적합한 식별자인가를 확인해야 한다
Monitor	이 지령은 **Monitor** 식별자의 이름을 정의한다. 이것이 Mointor 에 적합한 식별자인가를 확인해야 한다.
DefaultDepth	이 지령은 **Display** 항목에서 **Depth** 지령이 정의되지 않았을 때 쓰이는 색계조를 정의한다.

Display

Screen 항목에는 Display 라고 하는 보조항목이 있다. 여러가지 비데오방식들을 정의하는 Display 보조항목들을 다중으로 리용할수 있다.

앞의 실례로부터 첫번째 보조항목을 해석해 보자.

```
Subsection    "Display"
  Depth         24
  Modes         "1024x768"
  ViewPort   0 0
EndSubsection
```

이 Display 보조항목은 4 개의 지령들을 정의하는데 그것을 표 12-10 에 주었다.

표 12-10. Display 보조항목지령

지 령	효 과
Depth	이 지령은 화소당 비트수로 현시장치의 색계조를 정의한다. 실례로 8bit 는 2^8 혹은 256 색, 16bit 는 2^{16} 혹은 65536 색, 24bit 는 2^{24} 혹은 16.7×10^6 색을 허용한다.
Modes	이 지령은 현시장치의 분해능을 정의한다. 일반적인 분해능들은 1024X768, 800X600, 640X480 이다.
ViewPort	이 지령은 가상탁상의 가능한 크기를 정의한다. 실례로 640X480 화면은 1024X768 가상현시구역에 포함될수 있다. 마우스가 화면의 경계에 닿으면 현시는 가상현시의 경계까지 흐르게 할것이다. **ViewPort** 지령은 화소단위로 수직탁상의 수평수직치수를 정의하는데 x 에 의해서가 아니라 공백에 의해서 값들을 분리시킨다. **ViewPort 0 0** 은 가상탁상이 없다는것을 표시한다.
Virtual	이 지령은 **Modes** 지령보다 더 큰 화면크기 혹은 현시장치의 분해능을 정의한다. 이 지령을 주면 마우스나 지시장치를 리용하여 더 커진 가상구역으로 옮겨 갈수 있다.

ServerLayout

ServerLayout 항목은 Screen 과 InputDevice 항목을 결합함으로써 구성을 완성한다. 이미 론의된바와 같이 Screen 항목은 Monitor 와 Device 항목의 정보를 결합시킨다. ServerLayout 항목은 《표준》구성을 요구하지 않는다. 그것이 XF86Config 파일의 한 부분이 아니면 실제적인 Screen 과 건반마우스 InputDevice 가 리용된다.

다음의 실례에서 ServerLayout 항목을 설명한다.

```
Section "ServerLayout"
#   Identifier    "Configuration 1"

#   Screen        "Sony Setup"
#   Screen        "Samsung Setup" RightOf "Sony Setup"

#   InputDevice   "Mousel"          "CorePointer"
#   InputDevice   "Keyboard1""CoreKeyboard"

EndSection
```

이 ServerLayout 항목은 표 12-11 에서와 같은 세개의 지령을 지정한다.

표 12-11. ServerLayout 항목지령

지 령	효 과
Identifier	이 **ServerLayout** 의 유일한 이름을 지정한다.
Screen	리용하려는 **Screen** 항목들을 지정한다. 이 경우에 **Identifier** 지령은 "**Sony Setup**"와 "**Samsung Setup**"으로 된다.
InputDevice	리용하려는 **InputDevice** 항목을 지정한다. 이 경우에 두개의 **InputDevice** 항목이 있게 되는데 그것들의 **Identifier** 지령으로 는 "**Mouse1**"과 "**Keyboard1**"이다.

X Windows 의 기동순서

가장 좋은 현시질을 제공하기 위하여 XFree86 봉사기를 구성하는것외에 X Windows 환경을 기동하는 경로를 구성할수 있다. 여기에는 매 사용자가 자기의 X Windows 기동순서를 조종할수 있게 해주는 두개의 기본파일인 .xinitrc 와 .Xclients 가 있다. 이 두개의 파일은 사용자의 홈등록부에 있다. 설치하면 체계기정파일을 태우는데 Red Hat Linux 7.1 에 있는것은 /etc/X11/xinit/xinitrc 와 /etc/X11/xinit/Xclients 이다.

.xinitrc 파일

xinit 는 X 봉사기와 초기의뢰자프로그람, 보통 창문관리기를 기동할 때 리용되는 특수한 프로그람이다. 기정으로 startx 는 먼저 사용자의 홈등록부에 .xinitrc 파일이 존재

하는가를 검사하고 이 파일에 기초하여 xinit 파일을 실행한다. 사용자의 .xinitrc 파일을 찾지 못한 경우 xinit 는 체계에 있는 xinitrc 파일(Red Hat Linux 7.1 에서는 /etc/X11/xinit/xinitrc)을 리용한다. 만일 이것도 찾지 못하면 xinit 는 X 봉사기를 기동시킨후에 한개의 xterm 창문을 펼친다.

Xinitrc 나 .xinitrc 파일은 실행가능한 쉘스크립트이다. 제 16 장《쉘에 대한 리해》에서 쉘에 대하여 고찰한다. 그러므로 이 장에서는 쉘이나 쉘스크립트의 구체적인 내용보다 Red Hat Linux 7.1 로부터 기정 xinitrc 파일을 빨리 찾아 내는 수단을 제공한다(행번호들은 실지 파일의 부분이 아니다.).

```
1:   #!/bin/sh
2:   # (c) 1999, 2000 Red Hat, Inc.
3:   userresources=$HOME/.Xresources
4:   usermodmap=$HOME/.Xmodmap
5:   userxkbmap=$HOME/.Xkbmap

6:   sysresources=/etc/XII/Xresources
7:   sysmodmap=/etc/XII/Xmodmap
8:   sysxkbmap=/etc/XII/Xkbmap

9:   # merge in defaults
10:  if [ -f "$sysresources" ]; then
11:        xrdb -merge "$sysresources"
12:  fi

13:  if [ -f "$userresources" ]; then
14:      xrdb -merge "$userresources"
15:  fi

16:  # merge in keymaps
17:  if [ -f "$sysxkbmap" ]; then
18:      setxkbmap 'cat "$sysxkbmap"'
19:      XKB_IN_USE=yes
20:  fi

21:  if [ -f "$userxkbmap" ]; then
22:      setxkbmap 'cat "$userxkbmap"'
23:      XKB_IN_USE=yes
24:  fi

25:  if [ -z "$XKB_IN_USE" -a ! -L /etc/XII/X ]; then
26:      if grep '^exec.*/Xsun' /etc/XII/X > /dev/null 2>&1 && [ -f
               /etc/XII/XF86Config ]; then
27:      xkbsymbo1s='sed -n -e 's/^[  ]*XkbSymbo1s
               ]*"순(.*순)".*$/순1/P' /etc/XII/XF86Config'
```

```
28:        if [ -n "$xkbsymbols" ]; then
29:          setxkbmap -symbols "$xkbsymbols"
30:          XKB_IN_USE=yes
31:        fi
32:        fi
33:    fi

34:    # xkb and xmodmap don't play nice together
35:    if [ -z "$XKB_IN_USE" ]; then
36:        if [ -f "$sysmodmap" ]; then
37:          xmodmap "$sysmodmap"
38:        fi

39:        if [ -f "$usermodmap" ]; then
40:          xmodmap "$usermodmap"
41:        fi
42:    fi

43:    unset XKB_IN_USE

44:    # The user may have their own clients they want to run. If they
45:    # don't, fall back to system defaults.

46:    # run all system xinitrc shell scripts.
47:    for i in /etc/X11/xinit/xinitrc.d/* ; do
48:        if [ -x "$I" ]; then
49:    . "$i"
50:        fi
51:    done

52:      if [ -f $HOME/.Xclients ]; then
53:      exec $HOME/.Xclients
54:    elif [ -f /etc/XII/xinit/Xclients ]; then
55:        exec /etc/XII/xinit/Xclients
56:    else
57:            # failsafe settings. Although we should never get here
58:            # (we provide fallbacks in Xclients as well) it can't
                  hurt.
59:            xclock -geometry 100x100-5+5 &
60:            xterm -geometry 80x50-50+150 &
61:            if [ -f /usr/bin/netscape -a - f usr/doc/HTML/index.html
   ]; then
62:                  netscape /usr/doc/HTML/index.html &
```

```
63:          fi
64:          if [ -f /usr/X11R6/bin/fvwm2 ]; then
65:               exec fvwm2
66:          else
67:               exec twm
68:          fi
69:   fi
```

다음의 단계들을 거친다.

1. 3~8 행 스크립트전반에 필요한 파일들의 위치를 설정한다. 이것들은 다음행들에서 리용되는 변수이다.

주의 6 행에서 sysresources 로 표시된 System X Resources 파일은 Red Hat Linux 7.1 에 있는 **/etc/X11/xdm/Xresources** 파일에 배치된다. 후에 보게 되는것처럼 이 파일을 일반적인 기정 X Windows 우선권으로 설정하려면 실제적인 파일의 위치를 반영하는 6 행을 변경시켜야 한다.

2. 10~12 행 공용인 Xresources 파일이 존재하면 그것을 적용한다 (X 원천에 대한 설명은 다음절에서 본다.).
3. 13~15 행 사용자가 자기의 홈등록부에 .Xresources 파일을 가진다면 그것을 현행설정들에 병합하여 적용한다.
4. 17~20 행 전반적인 X Windows 건반배렬파일이 존재하면 그 파일에 있는 규칙들을 적용한다.
5. 21~24 행 사용자가 자기의 홈등록부에 X Windows 건반배렬파일을 가진다면 그것을 적용한다.
6. 25~33 행 이 행들은 대체로 체계에서 Sun X 봉사기를 리용하지 않는한 적용하지 못한다.
7. 34~43 행 사용자가 정의되거나 체계가 정의된 X Windows 건반배렬파일사이에 충돌이 있는 경우 그 파일에 건반정의를 설정한다.
8. 47~51 행 이 행들은 /etc/X11/xinit/xinitrc.d/등록부에 있는 모든 쉘스크립트들을 실행한다.
9. 52~69 행 이 부분에서는 X 봉사기가 기동한후에 실지로 첫 의뢰기를 기동시킨다는것을 보여 준다. 먼저 사용자가 .Xclients 파일을 가지는지 검사한다. 이 파일이 존재하면 파일은 거기에서 정의된 임의의 의뢰기들을 기동시킨다. 이 파일이 존재하지 않으면 Xclients 대역파일의 실체를 검사하고 그것이 존재하면 실행시킨다. 마지막으로 아무 파일도 존재하지 않으면 xclock, xterm 창문, Web 열람기를 비롯해서 몇개의 기정프로그람들을 기동시킨다. 그리고 가능하다면 fvwm2 혹은 twm 창문관리기를 기동시킬수도 있다.

Xclients 파일

xinitrc 파일설명에서 본것처럼 사용자는 자기의 홈등록부에 .Xclients 파일이 있으면 대역파일인 Xclients 파일을 무시할수 있다. 어느 경우에나 Red Hat 환경에서 xinit 는 X 봉사기가 기동된후에 초기의뢰기들을 기동시키기 위하여 임의의 한개 파일을 호출한다.

xinitrc 처럼 이 파일은 하나의 쉘스크립트언어이며 쉘스크립트언어의 모든 규칙들에 따른다. Xclients 가 리용할수 있는 기능들의 형식을 리해하기 위하여 이 절에서는 Red Hat Linux 7.1 을 적재한 기정 Xclients 파일을 고찰한다(행번호들이 다시 추가되였다). 즉

```
1:   #!/bin/bash
2:   # (c) 1999, 2000 Red Hat, Inc.

3:   # check to see if the user has a preferred desktop
4:   PREFERRED=
5:   if [ -f /etc/sysconfig/desktop ]; then
6:       if [ -n "'grep -i GNOME /etc/sysconfig/desktop'" ]; then
7:     PREFERRED=gnome-session
8:       elif [ -n "'grep -i KDE /etc/sysconfig/desktop'" ]; then
9:     PREFERRED=startkde
10:      elif [ -n "'grep -i AnotherLevel /etc/sysconfig/desktop'" ];
   then
11:  PREFERRED=AnotherLevel
12:      fi
13:  fi
14:  if [ -n "$PREFERRED" -a "$PREFERRED" != "AnotherLevel" ] && ⮐
15:    which $PREFERRED >/dev/nu11 2>&1; then
16:      PREFERRED='which $PREFERRED'
17:      exec $PREFERRED
18:  fi

19:  # now if we can reach here, either they want AnotherLevel or there
20:  # was no desktop file present and the PREFERRED variable is not set.

21:  if [ -z "$PREFERRED" ]; the

22:      GSESSION=gnome-session
23:      STARTKDE=startkde

24:      # by default, we run GNOME.
25:      if which $GSESSION >/dev/nu11 2>&1; then
26:        exec 'which $GSESSION'
27:      fi

28:      # if GNOME isn't installed, try KDE.
29:      if which $STARTKDE >/dev/nu11 2>&1; then
30:        exec 'which $STARTKDE'
31:      fi
32:  fi
```

```
33:  # Last, try AnotherLevel
34:  # these files are left sitting around by TheNextLevel.
35:  rm -f $HOME/Xrootenv.0
36:  rm -f /tmp/fvwmrc* 2>/dev/nu11

37:  # First thing - check the user preferences
38:  if [ -f $HOME/.wm_style ] ; then
39:      WMSTYLE=' cat $HOME/.wm_style |tr A-Z a-z'
40:      case "$WMSTYLE" in
41:    afterstep)
42:    exec /usr/X11R6/bin/RunWM - AfterStep
43:    ;;
44:    windowmaker|wmaker)
45:    exec /usr/X11R6/bin/RunWm - WindowMaker
46:    ;;
47:    fvwm95|fvwm|fvwm2)
48:    exec /usr/XIIR6/bin/RunWM -Fvwm95
49:     ;;
50:    mwm|1esstif)
51:    exec /usr/XIIR6/bin/RunWM -FvwmMWM
52:    ;;
53:      esac
54:  fi

55:  # Argh! Nothing good is installed. Fall back to fvwm2 (win95-style) or
     twm
56:  /usr/XIIR6/bin/RunWM -Fvwm95 || {
57:      # gosh, neither fvwm95 nor fvwm2 is available;
58:      # fall back to failsafe settings
59:      xclock -geometry 100x100-5+5 &
60:      xterm -geometry 80x50-50+150 &
61:      if [ -f /usr/bin/netscape -a -f
         /usr/share/doc/HTML/index.html ];then
62:    netscape /usr/share/doc/HTML/index.html &
63:      fi
64:      if [ -f /usr/X11R6/bin/fvwm ]; then
65:    exec fvwm
66:      else
67:    exec twm
68:      fi
69: }
```

Xclients 파일의 내용을 단계별로 갈라서 보자.

1. 4~13 행 요구하는 탁상을 설정한다. 제 11 장 《KDE 의 리용》에서 본바와 같이 4 행에서 REFERRED 변수를 리용하여 요구하는 탁상을 설정할수 있다. PREFERRED 이 어떤 탁상에도 설정되여 있지 않으면 5～12 행은 GNOME, KDE, AnotherLevel 의 /etc/sysconfig/desktop 파일에서 보여 준 기정탁상들을 검색한다.
2. 14~18 행 PREFERRED 탁상이 AnotherLevel 가 아닌가를 확인한다.
3. 21~32 행 PREFERRED 탁상이 없으면 GNOME 을 설정한다. GNOME 이 설치되여 있지 않으면 KDE 를 설정한다.
4. 35~36 행 X Windows 가 마지막에 남겨 놓은 림시파일들을 삭제한다.
5. 38~54 행 AnotherLevel 창문관리기의 형을 검색하고 그 형으로 Another Level 을 기동한 다음 스크립트를 끝낸다.
6. 56~69 행 Fvwm95 형으로 AnotherLevel 을 기동한다. 실패하면 xclock, xterm, Netscape 를 기동하고 fvwm 이 존재하면 그것을 기동시킨다. Fvwm 이 존재하지 않으면 대신 twm 을 기동한다.

X Windows 가 기동할 때마다 실행되는 의뢰기들을 추가하려면 자기의 홈등록부에 있는 .Xclients 파일에 Xclients 파일을 복사하고(파일복사에 대해서는 제 14 장 《파일다루기》에서 설명한다.) 그것을 편집한다. 이 처리에서 중요한것은 그 항목들이 있는 각이한 창문관리기들이 개시되기전에 실행하려는 프로그람들에 대한 지령을 추가하는것이다. 그 리유는 각이한 창문관리도구들을 개시할 때 쓰이는 exec 예약어가 창문관리도구가 개시되자마자 실행을 중지하는 스크립트이기때문이다.

X Resources

X 자원자료기지는 색, 서체 등과 같은 속성들을 조종하는 우선권정보를 가진 응용프로그람들을 제공한다. X 자원은 지령행기발로 조종할수 있는 동작의 거의 모든 양상을 조종하기 위하여 대부분 X Windows 응용프로그람으로 리용할수 있다. X 자원자료기지가 있으면 그 응용프로그람에 대해서 표준인것들보다 더 알맞는 기정값들을 정의할수 있다.

X Resources 의 동작방법

X Windows 를 기동할 때마다 각이한 설정들이 X 자원자료기지에 적재된다. 기정설정들은 xinitrc 에서 이미 론의된 각이한 파일에서 취한다. 그것들은 자기의 홈등록부에 있는 임의의 .Xdefaults 파일에 의하여 수정된다. 그러나 이 설정들은 세부적이고 복잡하다.

이 모든것들을 가지고 작업하려면 응용프로그람관련정보는 알맞는 응용프로그람들만이 X 자원항목들로부터 영향을 받도록 분류되여야 한다. 이것은 클라스별로 응용프로그람들을 갈라 놓음으로써 수행된다. 대부분 응용프로그람들은 클라스를 가진다. 응용프로

그람의 문서에서 클라스이름을 지정할수 있다. 실례로 xload 는 Xload 클라스에, xterm 은 Xterm 클라스에 속한다. 비슷한 응용프로그람들이 존재하는 곳에서 그것들은 종종 같은 클라스(oclock 와 xclock 는 Clock 클라스에 속한다)의 부분에 놓인다.

주의 응용프로그람과 클라스사이는 대문자를 써서 가른다. 실례로 xterm 은 표준 X Windows 말단응용프로그람이며 XTerm 은 그것이 속하는 클라스이다.

매개 클라스에 대해서 자원들의 표준을 설정하면 전경 및 배경색, 창문크기 및 위치, 표준서체와 같은 특성들을 지정할수 있다. 또한 관련된 자원들을 묶어 놓은 자원클라스들도 있다. 실례로 Foreground 클라스는 응용프로그람들이 가질수 있는 추가적인 전경과 관련된 자원은 물론 전정자원들도 포함한다. 보통 개별적인 자원들에 주의를 돌릴 필요는 없으며 자원클라스만으로 작업하게 된다.

X Resources 를 .Xdefaults 로 설정

설정하는 X 자원들은 X 자원자료기지에 입력된 항목을 포괄한다. 이 항목들은 다음과 같은 형태를 취한다.

```
<ApplicationClass>|<applicationName>*<ResourceClass>|resourceName> :<value>
```

Linux 의 man 페지들과 기타문서들에서 수직막대기(혹은 관문자:|)는 《혹은(or)》으로 많이 쓰인다. 웃실례에서 첫번째 내용은 <ApplicationClass> 혹은 <application Name>이고 두번째 내용은 <ResourceClass> 혹은 <resourceName>이다.

이 값들은 일반적으로 사용자정의자원을 위한 .Xdefaults 파일에 배치된다. 실례로 .Xdefaults 파일을 들어 보자.

```
XTerm*background: Black
XTerm*foreground: Wheat
XTerm*cursorColor: Orchid
Xterm*reverseVideo: false
Xterm*scrollBar: true
XTerm*reverseWrap: true
XTerin*font: fixed
XTerm*fu11Cursor: true
XTerm*scrollTtyOutput: off
XTerm*scrollKey: on
XTerm*tit1eBar: false
xclock*Geometry: 100+100+100+100
xclock*Foreground: purple
xclock*Background: mauve
```

이 .Xdefaults 파일은 Xterm 응용프로그람클라스와 xclock 응용프로그람에 대해서다 .X references 들을 설정한다. XTerm 클라스에 대해서 색과 창문속성(scrollbar 의 설치와 같은) 등을 설정하는 자원들을 조사한다. Xclock 에 대해서 색과 설치된 기하학적조항을 조사한다. 이 자원들에 의해 주어 지는 값은 -fg 와 -geometry 와 같은 지령행기발로 제공되거나 그 값들과 일치한다.

X Resources 의 자료기지

X 자원에 대한 원천들은 app-defaults 파일이다. X 와 KDE 응용프로그람을 위해 많은 app-defaults 파일들이 준비되여 있다. 자기의 .Xdefaults 파일에서 리용할수 있는 이 설정파일들을 볼수 있다. Red Hat Linux 7.1 에서 이 파일들은 /usr/X11R6/lib/X11/app-defaults/등록부에 배치되여 있다. KDE 응용프로그람에는 /usr/share/apps/kdisplay/app-defaults/등록부에서 지정한 app-defaults 파일들이 있다.

X References 를 적재하기 위한 xrdb 지령의 리용

보통 .Xdefaults 파일은 X Windows 가 .xinitrc 파일안에서 기동될 때 적재된다. 그러나 새 값은 X Windows 가 xrdb 지령으로 기동하는 과정에 자료기지에 적재될수 있다. 이것은 요구하는 결합을 정확히 조사할 때까지 값들에 대해서 실험할 때 특별히 리용할수 있다.

실험적인 설정값을 Experiment 라는 이름을 가진 파일에서 설정했다면 다음의 지령으로 이러한 설정들을 자료기지에 림시로 적재할수 있다.

```
$ xrdb -merge ecpriment
```

일단 변경이 만족하게 되였으면 experiment 파일부터 .Xdefaults 파일까지 설정들을 추가함으로써 X Windows 대화조종에 그것들을 삽입시킬수 있다.

▶미리 알아두기

이 장에서는 주로 X Windows 에 중심을 두고 고찰하였다. 이제는 Linux 즉 지령행환경에 완전히 정통하기 위한 기본부분에로 넘어 갈수 있다.

제 13 장 《Linux 지령의 기본개념》와 제 14 장《파일다루기》에서는 기초적인 Linux 지령들을 고찰한다. 거기서는 디스크구동장치에서 파일과 등록부로 작업하는 방법과 기본지령들을 고찰한다. 장의 마감에 작업에 강력한 지령행환경을 보장해 주는 쉘에 대하여 배우며 Linux 체계구성을 리해하는데서의 어려움을 덜어 주는 도형방식 RedHat 조종도구를 고찰하게 된다.

제4편. 기본요점습득

제13장. Linux지령의 기본개념

이 장에서는 Unix지령행환경에 대해서 본다. 지령행환경은 Linux의 중심이며 그 동력으로 되는 부분이다. X Windows가 있으면 많은 사용자들이 도형방식응용프로그람을 재빨리 그리고 쉽게 만들수 있다는데 대해서는 이미 보았다. 그러나 Linux체계로 충분히 조작도 하고 작업도 할수 있는 능력이 아직 부족하다. 이러한 능력은 지령을 써보느라면 점차 알게 된다.

여기서는 ls, find, grep와 같은 가장 일반적인 지령들을 어떻게 리용하는가에 대하여 배우게 된다. 이런 지령들은 거의 대부분의 배포물에 들어 있기때문에 Unix사용자들은 이것들과 비슷한 지령들이 무엇을 할수 있고 능숙한 사용자들에게 어떤 힘을 안겨 줄수 있는가를 곧 알게 된다.

주의 이 장을 통해서 Linux지령행을 리용할줄 알아야 한다. 지령행은 두가지 방법으로 얻을수 있다. 하나는 체계를 기동할 때 하나의 Linux조종탁에 가입하는것이고 다른 하나는 X Windows로부터 xterm, GNOME말단 또는 KDE의 조종탁과 같은 말단창문을 기동시키는것이다.

Linux지령이란 무엇인가

개별적인 지령들을 보기에 앞서 《지령》이라는 말이 무엇을 의미하는가에 대하여 정확히 리해하여야 한다.

DOS환경에서 작업해 본 사용자들은 DIR, COPY 및 ATTRIB와 같은 조작체계의 기본특성을 포함하는 지령의 개념을 잘 알고 있다. 이 지령들은 보다 복잡한 기능동작을 수행할수 있고 정교한 묶음파일을 작성할수 있는 기초를 준다.

그러나 다른 조작체계에서와 마찬가지로 DOS세계에는 리용할수 있는 지령들이 적고 일반적으로는 사용자가 새로운 지령을 첨가할수 없는 이른바 《정적》인것으로 되여 있다.

Unix세계에서는 Linux의 확장에 의하여 그 개념이 달라 진다. 지령이란 실행할수 있는 어떤 파일이다. 즉 지령은 자료나 구성정보를 포함한 파일과 달리 실행할수 있게 설계된 파일들로 구성된다. 이것은 실행할수 있는 어떤 파일이 체계에 첨가되면 그 파일은 주어 진 체계에서 새로운 지령으로 된다는것을 의미한다.

Linux지령의 실행

X Windows응용프로그람의 기동과정에 대하여 보았으므로 아마 지령실행에 대한 어느 정도의 리해를 가졌을것이다. 지령재촉문에 지령이름을 간단히 입력한다. 즉

$ *command*

만일 지령이 자기 경로에 없다면 완전한 경로와 지령이름을 입력해야 한다. 즉

```
$ /usr/bin/command
```

Linux에서 경로의 리용법

경로에 대한 개념은 약간의 설명을 요구한다. 가입할 때 매개 사용자는 기정경로를 가진다. 다음과 같은 지령으로 자기의 기정경로를 찾는다.

```
$ echo $PATH
```

이때 출력으로는 다음과 같은것이 나타난다.

```
$ usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/home/mj/bin
```

$PATH는 두점으로 분리된 등록부의 목록이다. 만일 지령이 경로가 없이 입력되면 지령과 관련된 파일들에 대하여 경로에 있는 모든 등록부가 차례로 검토된다. 우의 실례에서 **guess**라는 이름을 가진 지령이 **/usr/local/bin**과 **/home/mj/bin**의 량쪽 등록부에 있다면 **/usr/local/bin**안에 있는것이 실행된다. 만일 **/home/mj/bin**등록부안에 있는 **guess**지령을 리용하려면 완전한 경로를 리용해야 한다. 즉 **/home/mj/bin/guess**지령을 써야 한다. 제16장 《쉘에 대한 리해》에서 경로설정방법을 구체적으로 보게 된다.

잇달아 련결된 다중지령(관련결이라고 한다)을 포함하는 보다 복잡한 지령실행방법에 대해서는 제16장에서 보게 된다.

공통적인 Linux지령

Red Hat Linux 7.1과 같은 공통적인 Linux배포물에서 리용하는 Linux지령의 수는 대단히 많다. 그러나 보통 수준 높은 사용자라고 해도 이 지령들가운데서 불과 얼마 안되는 선택된 지령들만을 리용하게 된다.

이 절에서 나오는것은 가장 흔히 쓰이는 Linux지령들이다. 이 지령들은 체계에서 무엇이 진행되고 있는가를 조사하기 위하여 등록부의 옮기기로부터 잃어 버린 파일의 조사에 이르기까지의 과제범위를 포괄한다. 이 절에서 보게 되는 지령들은 다음과 같다.

- su
- pwd, cd 및 ls
- more 및 less
- find, locate, whereis 및 grep
- tar 및 gzip
- man

su

su지령은 가장 기초적이며 많은 각이한 과제처리에서 쓸모 있는 지령이다. su지령은 보통 서로 다른 사용자들사이에서 절환용으로 쓰인다. user1로 가입하였는데 어떤 작업을 수행하기 위하여 user2로 절환하였다가 다시 user1로 작업하기 위해 반대로 절환하려고 하는 경우를 보자.

탈퇴하고 user2로 가입하여 작업을 진행하고 다시 탈퇴하여 user1로 돌아 간다. 그러나 이 과정에는 약간한 시간소비가 있을수 있다. 하나의 가상조종탁에서는 user1로 가입하고 다른데서는 user2로 가입하며 절환기는 앞뒤로 절환된다. 이런 문제는 서로 다른 화면사이의 절환과 같은 작업을 위해 필요하다.

세번째 선택항목은 su지령을 리용하기 위한것이다. 만일 user1로 가입했다가 user2로 가입하려면 간단히 다음과 같이 입력해야 한다.

```
$ su user2
```

그러면 user2에 대한 통과암호를 재촉한다. 즉

```
$ su user2
password:
```

작업이 끝난 다음 user1로 돌아 가려면 exit지령을 입력해야 한다. 즉

```
$ exit
```

이것을 종합하면 완전한 대화조종은 다음과 같다.

```
[user1@localhost user1]$ su suer2
Password:
[user2@1oca1host userl]$ some commands
[user2@1ocalhost userl]$ exit
exit
[userl@loca1host userl]$
```

su지령은 상급사용자라고 하는 뿌리사용자가 되여야 할 때, 많은 관리과제를 수행하려고 할 때, 사용자회계를 만들고 경영하는것을 비롯하여 망의 구성과 인쇄기의 구성을 진행할 때 등에 공통적으로 쓰인다.

만일 사용자이름이 없이 su지령을 주면 뿌리통과암호에 대한 재촉문이 나타나고 일단 이것을 제공하면 뿌리사용자로 동작하도록 절환된다. 즉

```
[user1@localhost user1]$ su
password:
[root@localhost user1]#
```

만일 뿌리사용자로 가입되면 통과암호없이 체계에서 임의의 사용자가 되도록 su를 리용할수 있다(그러므로 뿌리통과암호를 노리는 자들로부터의 기밀을 담보하는 중요한 문제가 해결된다.). 이것은 오유수정문제때문에 각이한 사용자가 되여야 하지만 다른 사용자의 통과암호를 반드시 알 필요가 없는 체계관리기에게 쓸모 있는것이다.

다음실례에서 뿌리사용자별명지령이 나타날 때 현시장치에 통과암호재촉문이 나타나지 않고 user1이 되기 위하여 su를 어떻게 리용하는가에 대하여 주의를 돌려야 한다. 즉

```
[root@localhost /root]#su user1
[user1@localhost /root]$
```

su지령은 그밖에도 많은 쓸모 있는 특성들을 가지고 있으므로 발전된 체계관리과제들에서도 자주 쓰인다. 그런 내용에 대해서는 **su** 사용설명서페지에서 볼수 있다. **su** 사용설명서페지를 보는 방법을 알려면 이 장의 뒤부분에 있는 man지령에 대한 부분을 보아야 한다.

pwd, cd 및 ls

pwd, cd 및 ls지령들을 묶으면 등록부나 파일과 작업하기 위해 필요한 기초도구가 갖추어 진다.

이 세개의 지령가운데서도 현재 작업중에 있는 등록부용으로 쓰이는 pwd지령이 가장 기본적인것이다. 이 지령을 입력하고 Enter건을 누르면 현재 작업중에 있는 등록부가 나타난다. 즉

```
$ pwd
/home/armand
```

이 경우에 pwd지령은 그에 대한 대답으로 /home/armand로 화면을 복귀시킨다. 이것은 사용자 armand의 홈등록부에 사용자가 있다는것을 말해 준다.

cd지령이 있으면 대상의 현행상태를 보다 간단히 조사할수 있다. 이것은 실제적으로 상태절환도 한다. cd지령은 현행등록부를 체계에 있는 다른 가능한 등록부에로 절환시킬수 있게 한다.

앞에서 본 현행등록부가 /home/armand인 실례를 보기로 하자. cd지령을 리용해서 armand홈등록부의 wordfiles라고 하는 보조등록부에로 절환할수 있다. 즉

```
$ cd wordfiles
$ pwd
/home/armand/wordfiles
```

등록부를 변경한 다음 pwd지령을 입력하면 그것으로 끝내도록 한다.

마찬가지로 다음과 같은 지령으로 /tmp라는 체계의 림시등록부로 절환할수 있다.

```
$ cd /tmp
$ pwd
/tmp
```

차이는 등록부의 앞에 놓인 빗선(/)에 있다. 첫번째 지령 cd wordfiles에는 빗선이 없다. 따라서 변환은 현행등록부에 관계된다. 만일 Mike의 홈등록부 /home/mike로부터 같은 지령을 실행하면 /home/mike/wordfiles등록부에서 끝나게 된다.

두번째 지령 cd /tmp에는 빗선이 있다. 이 두번째 지령은 임의의 등록부로부터 기동할수 있으며 같은 장소에서 끝나게 된다.

마지막으로 ls지령은 현행등록부를 조사하는데 리용될수 있다. 가령 armand홈등록부의 내용을 조사하기 위하여 ls지령을 주면 다음과 같이 된다.

```
$ ls
2341ch11a.doc                  dead.letter    scmp-jpc.bak
DISKCOPY.COM                   foo            svgalib-1.2.11-4.i386.rpm
```

```
DRWEBDEM.IMG                       foo.html      test.txt
Xconfigurator-3.26-1.i386.rpm      mail          wabi
Xrootenv                           nsmail        xserver-1.1-1.i386.rpm
armand                             scmp-jpc
```

파일이나 등록부의 이름이 여러 렬로 현시된다는것과 렬의 너비는 가장 긴 이름의 너비에 의하여 결정된다는것을 명심하여야 한다.

현행등록부의 내용을 목록화할뿐아니라 체계에서 임의로 호출할수 있는 등록부의 내용도 목록화할수 있다. 가령 등록부 /usr의 내용을 목록화하려면 지령 ls /usr를 입력해야 한다.

```
$ ls /usr
XII       doc              i 486-1inuxaout    1ib        sbin
X11R6     dt               ibase              libexec    share
X386      etc              include            local      spool
bin       games            info               man        src
dict      i486-1inux-1ibc5 interbase          openwin    tmp
```

우에서 이야기한바와 같이 목록화될 때 렬의 너비와 수는 표시하여야 할 가장 긴 이름의 길이에 관계된다.

물론 실제로 이런 목록을 리용하는지 좀 의문이 될수 있다. 여기에는 파일을 가리키는 이름과 부분등록부를 가리키는 이름 그리고 파일의 크기가 얼마인가를 가리키는 정보가 없다.

그런 정보는 ls지령의 확장형인 ls -1(이 지령의 구조를 알려면 제14장《파일다루기》에 있는 ls지령에 대한 완전한 해설을 참고해야 한다.)을 리용해서 얻을수 있다. armand홈등록부에서 ls -1을 쓰면 다음과 같은 결과가 얻어 진다.

```
$ ls -1
total 1807
-rw-r—r—   1 armand   armand     52224 Apr   24 23:00   2341chlla.doc
-rw-r—r—   1 armand   armand     24325 May    9 16:06   DISKCOPY.COM
-rw-r—r—   1 armand   armand   1474979 May    9 16:06   DRWEBDEM.IMG
-rw-r—r—   1 armand   armand     52313 Jan   21 18:04   Xconfigurator
-3.26-l.i386.rpm
-rw-r—r—   1 armand   armand       396 May   19 23:09   Xrootenv
drwx------ 2 armand   armand      1024 May   1709:55    armand
-rw———     1 armand   armand     10572 May 18 22:29     dead.letter
-rw------- 1 armand   root        1455 Apr   24 21:38   foo
-rw-r—r—   1 armand   armand      2646 May    7 07:32   foo.html
drwx------ 2 armand   armand      1024 Dun    4 07:12   mail
drwx------ 2 armand   armand      1024 May   17 09:56 nsmail
-rw-r--r—  1 armand   armand      4288 May   14 22:17 scmp-jpc
```

```
-rw-r--r--    1 armand    armand          4289 May   14 22:12 scmp-jpc.bak
-rw-r--r—   1 armand     armand        95341 Mar   25 17:32 svgalib-1.2.11
-4.i386.rpm
-rw-rw-r--     1 armand     armand           94 May   17 11:44 test.txt
drwxr-xr-x   5 armand     armand        1024 May   19 23:07 wabi
-rw-r--r--   1 armand     armand          4493 Feb     4 15:31 xserver-wrapper-
1.1-l.i386.rpm
```

매개 파일에는 참고정보가 포함되여 있다는것을 명심해야 한다. 매개 파일이름의 왼쪽에 직접값으로 주어 진것이 가장 중요한 정보부분인데 파일이나 등록부의 마지막변경부와 같은것이 여기에 속하며 파일인 경우에는 날자의 왼쪽에 파일의 크기가 주어 진다(가령 1024-byte파일이면 1KB파일을 의미한다).

또한 매개 행의 맨 왼쪽에서 등록부를 문자d로 그리고 파일을 보통 짧은 이음표(-)로 표시한다는것을 알아야 한다. 가령 scmp-jpc는 파일이다. 즉

```
-rw-r--r-- 1 armand armand    4288 May 14 22:17 scmp-jpc
```

한편 mail은 등록부이다. 즉

```
drwx------  2 armand armand   1024 jun   4 07:12  mail
```

more 및 less

more와 less지령은 밀접히 련관되여 있으며 비슷한 기능을 가지고 있다. 그러면서도 흥미 있는것은 less의 능력이 more보다 더 크다는것이다.

이 두 지령의 기본목적은 긴 파일이나 본문목록을 한 화면 또는 창문에 단번에 표시하는데 있다. 이때 사용자는 본문을 훑어 보면서 페지를 아래로 이동할수도 있고 어떤 경우에는 본문을 훑어 보면서 반대로 이동할수도 있다. 이 두 지령은 현시된 본문을 탐색하는 능력도 가지고 있다.

xedit와 같은 편집프로그람을 열지 않고 긴 본문파일을 재빨리 훑어 보려고 할 때라든가 특별히 긴 등록부의 목록을 보려고 할 때를 비롯하여 많은 경우에 이 지령들이 쓸모 있다.

more

more지령부터 보기로 하자. more지령은 주어 진 본문을 훑어 보는 경우 규모가 큰 본문을 따라 가면서 한꺼번에 한행 또는 한 화면씩 앞으로 움직여 주는 가장 기본적인 지령이다.

가령 text file이라고 불리우는 큰 본문파일을 움직일 때 다음과 같은 지령을 주면 한번에 한페지를 볼수 있다.

```
$ more textfile
```

Enter건을 누르면 화면의 마지막행에 –More-(xy%)로 표시되는 본문과 함께 파일의 첫 화면이 펼쳐 진다. 여기서 xy는 화면에 나온 파일의 퍼센트를 나타낸다. 공백건을 누르면 옹근 한 화면분만큼 앞으로 건너 뛸수 있고 Enter건을 누르면 한번에 한행씩 앞으로 움직인다. 본문의 마감에 가닿으면 지령재촉상태로 돌아 간다.

파일을 앞방향으로 탐색하려면 단어나 문구단위로 탐색하려는것만큼 빗선(/)을 입력하고 그다음 Enter건을 누른다. 표시는 탐색되는 단어나 문구가 첫번째로 나타나는 곳까지 앞으로 건너 뛴다. 그리고 화면의 꼭대기에 탐색내용을 표시한다. 첫번째 탐색후에도 n을 입력하는데 따라 같은 탐색을 반복한다. 그러나 이때는 같은 단어나 문구로 돌아가므로 건입력하지 않아도 된다.

파일의 내용을 보기 위하여 more를 주는것외에도 관련결기능(구체적인 수법은 제16장에서 본다.)을 리용하면 more에 다른 지령의 결과도 통과시킬수 있다.

가령 한 체계에서 237행 또는 그이상의 행들에 결과를 만드는 /tmp라는 등록부의 내용을 보기 위하여 ls -1을 주면 가장 큰 xterm창문을 표시할수 있다. ls -1지령의 결과를 단번에 한 창문에서 보기 위해서는 지령결과를 more에 통과시켜야 한다. 즉

```
$ ls -1 /tmp | more
```

이 지령은 ls -1지령을 관기호(보통 대부분의 미국식건반에서 거꿀빗선우에 있는 2중수직두점이 대응된다.)로 more지령과 련결한다. 관기호를 쓰는것은 바로 《관련결》기능을 리용하는것으로 된다. 이 관잇기의 결과로 more가 ls -1지령의 결과를 현시하는데 리용되게 된다. 그것은 이미 앞에서 파일의 내용에 대하여 진행한것과 꼭같이 단번에 한 화면 또는 한행 아래로 이동시키거나 결과를 탐색한다는것을 의미한다.

주 의	more를 리용하기 위한 완전한 방법은 **more** 사용설명서페지에서 볼수 있다. 사용설명서페지의 리용에 대해서는 이 장의 뒤에 있는 man지령부분에서 보게 된다.

Less

사실 less지령은 more지령을 크게 개선한것이라고 볼수 있다. 이 지령은 앞에서 본 기본기능(한 화면이나 한행을 단번에 앞으로 움직이는것과 탐색하는것)밖에도 본문의 본론에서 수행할수 있는 다음과 같은 몇가지 보충적인 기능을 가지고 있다.

- 어떤 행으로 직접 건너뛰기
- 어떤 파일의 시작이나 끝으로 직접 건너뛰기
- 파일을 반대로 이동
- 파일을 반대로 탐색

어떤 행이나 파일에로 직접 건너뛰기 위해서는 행번호뒤에 문자 **g**를 입력해야 한다. 만일 행번호를 일일이 지적하지 않고 그냥 **g**를 입력하면 파일의 첫행으로 건너 뛰게 된다. 그러나 대문자 G를 입력하면 지적된 행번호가 없는 경우와 거의 같이 동작하는데 본문본론의 마지막행으로 건너 뛴다.

less를 써서 파일을 반대로 이동시키기는 쉽다. 웃방향건은 본문 한행을 단번에 우로 움직이며 아래방향건은 단번에 한행 아래로 움직인다. 단번에 한 화면을 반대로 넘기려면 Ctrl+B를 눌러야 한다.

마지막으로 빗선(/)뒤에 단어나 문구를 입력하고 Enter건을 누르면 현시하려는 본문을 앞방향으로 탐색하며 물음표(?)뒤에 단어나 문구를 입력하고 Enter건을 누르면 본문을 반대방향으로 탐색한다.

주 의	알고 있는바와 같이 이 지령들은 vi본문편집기에서의 선택항목에 따른다.

more지령에서와 같이 이 지령의 모든 자세한 특성에 대해서는 less사용설명서페지에서 볼수 있다. 사용설명서페지의 리용은 이 장의 마지막에 있는 man지령에 대한 부분에서 보게 된다.

find, locate, whereis 및 grep

find, locate, whereis 및 grep지령은 강력한 파일탐색지령이다. 이 지령들을 설명하자면 이 장을 모두 바쳐야 할 정도이므로 여기서는 가장 기본적인것에 대해서만 보기로 하자. 만일 지령들에 대한 완전한 설명을 바란다면 이 장의 마지막에서 보게 되는 man지령을 리용하는 방법을 배워서 사용설명서페지를 보아야 한다.

이 지령들은 모두 탐색용으로 쓰이지만 그것들의 목적은 다르다. find가 참조사항의 이름이나 자료를 비롯한 몇가지 기준에 따라 파일을 탐색한다면 grep는 파일의 내용을 탐색하는데 쓰인다.

find

이 책을 읽기전부터 콤퓨터를 다루고 있었다면 아마 어떤 파일을 새로 만들었는데 그것을 어디에 보관했는지 기억되지 않는 정황에 부딪칠 때도 있었을것이다. 이런 딱한 경우에 대답을 주는것이 바로 Unix의 find지령이다.

이 지령은 이름, 작성날자 또는 변경날자, 소유자(보통 파일을 만든 사용자), 파일보기 및 파일의 구체적류형 등으로 파일을 탐색하는데 쓰인다. 여기서는 가장 널리 쓰이는 이름에 의한 파일탐색에 대하여 설명한다.

find지령의 기본구조는 다음과 같다.

```
$ find strating-directory parameters actions
```

시작등록부는 탐색을 시작하는 장소를 지적한다. 가령 /home라고 지적하면 /home이라는 보조등록부만이 탐색된다는것을 의미하며(다시말하여 사용자의 홈등록부만이 탐색된다.) / means라고 지적하면 모든것이 탐색된다.

파라메터들은 탐색에 쓰이는 기준을 지적한다.이 경우에 탐색하려는 파일의 이름을 정의하기 위해서 지령 **switch**와 **argument -name 파일이름**을 리용한다.

작용항목은 탐색된 파일이 어떤 작용을 하는가를 지시한다. 보통 –print작용을 리용하게 되는데 그것은 현시될 파일의 완전한 이름과 경로를 지시한다. 이밖에도 find지령은 지시된 탐색을 수행하지만 목적에 맞지 않는 결과는 현시하지 않을수도 있다.

마지막으로 체계에 있는 foo라는 이름을 가진 모든 파일에 대하여 탐색하려는 경우에는 다음과 같은 지령을 주어야 한다.

```
$ find / -name foo -print
```

이런 경우에 결과는 다음과 같이 될수 있다.

```
$ find / -name foo –print
/tmp/foo
/name/armand/foo
/home/tdanesh/foo
```

일러두기	앞의 지령에서는 체계전체를 탐색하려고 한다는것을 명심해야 한다. 이것을 효과적으로 수행하자면 체계의 모든 등록부를 호출할수 있도록 뿌리사용자로 가입해야 한다. 만일 그렇지 않으면 find가 허가를 못 가진 등록부를 탐색하려고 시도할 때마다 매번 permission denied(허가거절)오유가 나오게 된다.

부분적인 파일이름에 대해서도 역시 탐색할수 있다. 가령 fo로 시작된 파일을 찾고 있다면 fo로 시작되고 임의의 조합으로 끝나는 모든 파일을 가리키기 위하여 fo*라는 표현을 쓸수 있다. 즉

```
$ find / -name 'fo*' –print
/tmp/foo
/var/1ib/texmf/fonts
/usr/bin/font2c
/usr/bin/mh/folders
/usr/bin/mh/folder
/usr/bin/mh/forw
/usr/bin/formail
/usr/bin/fontexport
/usr/bin/fontimport
/usr/bin/fold
etc.
```

fo*의 앞뒤에 단일인용부호를 주의해서 써야 한다. *기호를 쓸 때 전체 식의 앞뒤에 단일인용부호를 쓰지 않으면 find는 오유를 내보낸다. 즉

```
$ find / -name fo* -print -mount
find: paths must precede expression
Usage: find [path...] [expression]
```

만일 find지령으로 얻어 지는 결과가 한 화면을 넘으면 앞에서 ls -1지령으로 한것처럼 관련결기호와 more지령을 쓸수 있다. 즉

```
$ find / -name 'fo*'  -print  |  more
```

locate

find지령이 지나치게 길어 지면 대신에 locate지령을 쓸수 있다. 이 지령은 밤마다 작성되는 체계의 파일자료기지를 탐색한다. 이것은 find지령과 약간 다르게 동작하므로 자기의 이름에 있는 탐색문자렬을 조사하는 매개 파일과 등록부를 복귀시킨다. 가령 locate xauth 라는 지령은 다음과 같은 결과를 준다.

```
$ locate xauth
/home/mj/.xauth
/home/mj/.xauth/refcount
/home/mj/.xauth/refcount/root
/home/mj/.xauth/refcount/root/testlinux
/lib/security/pam_xauth.so
```

/usr/XllR6/bin/mkxauth
/usr/XllR6/bin/xauth
/usr/XllR6/man/manl/mkxauth.lx.gz
/usr/XllR6/man/manl/xauth.lx.gz
/usr/share/doc/pam-0.72/txts/README.pam_xauth
/usr/share/man/man8/pam_xauth.8.gz

이 지령이 /home/mj/.xauth 등록부와 /usr/X11R6/bin/mkxauth 그리고 /usr/X11R6/bin/xauth 지령을 비롯하여 본문문자렬 "xauth"를 포함하는 모든 파일들과 등록부들에 완전한 경로를 주는 방법을 보자.

이 지령은 find지령에 비하여 훨씬 빨리 동작한다. 결함은 locate지령이 보통 24시간에 한번씩만 갱신되는 자료기지파일에 기초하여 동작한다는것이다. 그러므로 지령의 결과에는 현행장소 지어는 방금 이동하거나 작성된 파일의 존재조차도 반영되지 않을수 있다.

whereis

알려 진 지령을 찾는 경우에는 지령, 원천코드 및 련관된 사용설명서페지로 등록부를 찾는데 whereis지령을 쓸수 있다. find나 locate지령과 달리 지령의 맞춤법이 정확해야 한다. 가령 whereis fdisk지령을 주면 다음과 같은 문장이 나탄나다.

```
$ whereis fdisk
fdisk: /sbin/fdisk /usr/share/man/man8/fdisk.8.gz
```

이 출력은 그것의 사용설명서페지에 따르는 fdisk지령(/sbin/fdisk)의 장소를 지정한다. fdisk원천코드의 위치가 보이지 않으므로 사용자는 그것이 설치되지 않은것으로 생각할수 있다.

whereis지령의 결함은 지정된 등록부묶음을 통해서 보아야 하기때문에 새로운 지령이나 방금 설치된 사용설명서페지를 찾을수 없는것이다.

grep

find는 그의 이름, 류형 또는 날자에 의하여 파일을 탐색하고 locate는 파일자료기지를 훑어 보고 탐색하며 whereis는 지령의 장소를 확정한다면 grep는 파일들에 있는 본문의 특수한 패턴에 대한 존재를 찾기 위하여 하나이상의 파일들의 내용을 잠간 훑어보는데 리용된다.

실례를 들어 보자. 즉 단어 radio를 포함하는 본문파일을 만들었고 홈등록부에 그것을 보관하였다는것을 알고 있다. 파일의 이름을 잊어 버렸지만 radio가 포함된 파일을 빨리 찾으려고 하는 경우에 grep를 리용하는것이 적합하다.

홈등록부에 그것이 있다고 보면 다음의 지령들은 홈등록부의 매 파일에서 단어 radio를 탐색하고 다음과 같은 결과를 내보낸다.

```
$ grep radio *
ab.txt:This is a test of searching for the word radio.
pop.txt:On another radio station, he found that
```

파일안에서 단어 radio가 나올 때마다 grep지령이 한행을 복귀한다는것을 명심해야

한다. 파일이름이 제시되고 계속해서 두점이 놓이며 그 뒤에는 단어가 나타날 행의 완전한 본문이 놓인다.

grep지령의 일반적인 패턴은 다음과 같다.

```
$ grep text-pattern file-list
```

본문패턴은 간단한 단어나 문구 또는 보다 완전한 정규적인 표현식으로 될수 있다(본문패턴탐색의 강력한 방법인 grep에 대한 정규적인 표현식의 리용법은 grep사용설명서페지에서 볼수 있다.). 파일목록은 지령해석기가 인정하는 임의의 형태를 가질수 있다. 파일목록을 이루는 표현식의 형태에 대한 완전한 설명에 대해서는 제16장을 참고해야 한다.

그러나 일반적으로 다음과 같은 형태를 가진 단일파일의 내용을 검사하든가

```
$ grep text-pattern file-name
```

파일의 내용을 검색하기 위해서 다음과 같은 지령을 리용할수 있다.

```
$ grep text-pattern *
```

여기서 별표는 검색되는 현행등록부의 모든 파일을 가리키는 표식이다.

본문패턴의 가장 간단한 형태는 한개 단어나 공백이 없는 부분적인 단어이다. 만일 “is a text”와 같은 문구를 탐색하려면 다음 실례에서와 같이 본문패턴을 인용괄호안에 넣어야 한다.

```
$ grep "is a test" *
ab.txt:This is a test of searching for the word radio.
```

more나 less지령을 비롯하여 지령의 결과를 관련결하는것이 때로는 아주 쓸모 있는데 이런 리치는 grep에서도 마찬가지이다. 현행등록부에서 May 12이라는 변경날자를 가진 모든 파일을 목록으로 만들려고 하는 경우를 보자. ls -1지령의 결과를 grep에 관련결하면 이 정보를 찾을수 있다.즉

```
$ ls –1 | grep "May 12"
-rw-r--r—     1 root     root      19197 May 12 21:17 rfbprotoheader.pdf
-rw-r—r--     1 root     root     110778 May 12 21:20 rfprotoA.zip
-rw-r--r—     1 root     root      17692 May 12 23:03 svnc-0.l.tar.gz
-rw-r—r--     1 root     root      25222 May 12 19:58 vnc
^»-3.3.1_javasrc.tgz
drwxr-xr-x    2 root     root       1024 May 12 21:49 vncjava
```

지령과 괄호

지령과 함께 쓸수 있는 괄호에는 세가지 형태의 괄호 즉 외괄호('), 2중괄호(") 및 거꿀외괄호(`)가 있다. 이것들사이의 차이점은 그것들이 date와 같은 매몰된 지령 및 $LOGNAME과 같은 변수에 어떤 영향을 주는가 하는ㄷ 있다. 일치되는 한쌍의 괄호들은 지령에 간결한 전체 표현을 보낸다. 실ㅔ로 $LOGNAME=mj이라고 하고 다음 지령들을 보자.

```
eth0 Welcome $LOGNAME, the date is date
eth0 'Welcome $LOGNAME, the date is date'
eth0 "Welcome $LOGNAME, the date is date"
```

eth0 "Welcome $LOGNAME, the date is `date`"

Linux는 괄호에 따라 date지령을 실행할수도 있고 못할수도 있으며 $LOGNAME을 변환할수도 있고 못할수도 있다. 이 지령들은 각이한 조합으로 된 괄호를 포괄하고 있으며 다음과 같은 결과를 내보낸다.

Welcome mj, the date is date
Welcome $LOGNAME, the date is date
Welcome mj, the date is date
Welcome mj, the date is Mon June 14 10:45:20 EDT 2001

첫번째 실례는 괄호가 없다. 그래서 $LOGNAME은 mj로 변환되지만 date지령은 실행되지 않는다. 두번째 실례는 외괄호를 쓰고 있으므로 $LOGNAME는 변환되지 않으며 date지령도 실행되지 않는다. 세번째 실례는 2중괄호를 쓰고 있다. 결과는 첫번째 지령에서와 같지만 2중괄호는 grep와 같은 지령과 함께 쓰면 쓸모가 있다. 마지막실례는 2중괄호안에 date를 둘러 싼 거꿀외괄호를 가지고 있다. 실례에서 볼수 있는바와 같이 Linux에서는 거꿀외괄호가 보조지령을 실행하게 한다.

tar 및 gzip

Windows 98/2000과 Mac OS를 비롯한 다른 OS를 리용하는 거의 모든 사용자들은 압축된 문서고의 개념을 잘 알고 있다. 압축된 문서고는 하나이상의 파일을 포함하고 있는 단일파일이다.

압축된 문서고는 인터네트에 공통적인 Zip파일을 가지는 경우와 같이 인터네트에 쏘프트웨어를 배포하기 위해서도 흔히 쓰인다. Linux는 Unix의 Zip문서고내용을 호출하는 unzip지령을 제공하며 쏘프트웨어를 배포하는데 보통 tar문서고가 리용된다. 이 문서고들은 gzip를 써서 압축을 진행하되 그 압축은 개별적인 파일에서 진행된다.

tar

tar프로그람은 처음에 테프에 체계여벌복사물을 만드는데 리용되였다(테프문서고라는 뜻에서 이름이 tar가 되였다). 현재는 배포물파일의 문서고를 만드는데 널리 쓰이고 있다.

tar문서고는 쉽게 만들수 있다. 즉

```
$ tar cvf tar-file-name file-list
```

이 지령은 파일이름 tar-file-name(보통 tar파일은 .tar라는 확장자를 가진다.)으로 지정되는 새로운 문서고를 만들며 이 문서고의 파일목록에 있는 모든 파일을 보관한다. 제16장에서 설명한것처럼 파일목록은 정확한 표현식의 결과로 생길수 있다. 문서고에 이 프로쎄스를 복사하는것을 명심해야 한다. 그래야 원본파일이 공정에서 없어 지는 위험을 면할수 있다.

tar지령뒤에는 련속선택항목이 직접 놓이게 된다는것을 명심해야 한다. 이 경우에 cvf가 놓인다. 매개 선택항목은 tar지령의 각이한 동작양상을 조종하기 위해서 리용한다.

- c는 문서고로부터 파일을 풀거나 현재 있는 문서고와 대조되는 문서고를 만든다는것을 나타낸다.
- v는 지령이 용장방식으로 실행되여야 한다는것을 가리킨다. 이것은 매개 파일이름이 문서고에 복사되는것처림 현시되여야 한다는것을 의미한다.
- f는 테프구동기와 대립되는것으로서 파일에 보관하고 있다는것을 의미한다.

례를 들어 현행등록부의 모든 .txt파일을 text.tar라고 하는 문서고에 보관하려면 다음의 지령을 주어야 한다.

```
$ tar cvf text.tar *.txt
ab.txt
pop.txt
```

파일이름이 문서고에 복사되는것처럼 그것들이 어떻게 목록화되는가 하는데 주의를 돌려야 한다.

때때로 등록부의 전체내용을 문서고에 복사해야 할 때가 있다. 다행히도 tar 등록부가 파일의 부분목록이면 모든 파일과 등록부의 보조등록부들을 문서고에 복사한다. 그러므로 vnc라고 하는 등록부가 있고 그 등록부의 전체 내용이 vnc.tar라고 하는 새로운 문서고에 복사되게 하려면 다음의 지령을 주어야 한다.

```
$ tar cvf vnc.tar vnc
```

그러면 다음과 같은 결과를 얻는다.

```
$ tar cvf vnc.tar vnc
vnc/
vnc/LICENSE.TXT
vnc/README
vnc/README.vncserver
vnc/Xvnc
vnc/classes/
vnc/classes/DesCipher.c1ass
vnc/classes/animatedMemoryImageSource.c1ass
vnc/cl asses/authenticationPanel.c1ass
vnc/classes/clipboardFrame.c1ass
vnc/classes/optionsFrame.c1ass
vnc/classes/rfbProto.c1ass
vnc/classes/vncCanvas.c1ass
vnc/classes/vncviewer.class
vnc/classes/vncviewer.jar
vnc/vncpasswd
vnc/vncserver
vnc/vncviewer
```

첫 행은 문서고에서 vnc등록부가 작성하고 그 등록부의 파일을 문서고에로 복사하는것을 가리킨다는것에 주목을 돌려야 한다.

만일 현행문서고를 가지고 있다면 보통 파일내용의 목록을 조사하든가 파일내용을 발취하든가 하는 어느 한쪽을 선택할수 있다.

문서고의 내용을 조사하기 위해서는 선택항목 c를 t로 바꾸어야 한다. 그다음 작성된 vnc.tar문서고의 파일목록을 보기 위하여 다음과 같은 지령을 리용할수 있다.

```
$ tar tvf vnc.tar
drwxr-xr-x root/root            0 2001-05-16 23:55 vnc/
-rw-r—r— root/root        18000 2001-01-23 16:52 vnc/LICENCE.TXT
-rw-r—r— root/root         6142 2001-01-23 16:53 vnc/README
-r--r--r---   root/root      601 2001-01-23 16:28 vnc/README.vncserver
-r-xr-xr-x   root/root  1286834 2001-01-23 13:00 vnc/Xvnc
drwxr-sr-x root/root            0 2001-01-23 16:24 vnc/classes/
-r—r—r— root/root          7143 2001-01-23 16:24 vnc/classes/DesCipher
.class
-r--r--r~   root/root       1329 2001-01-23 16:24 vnc/classes/
animatedMemoryImageSource.c1ass
-r--r--r-- root/root         2068 2001-01-23 16:24 vnc/classes/
authenticationPanel.class
-r--r--r-- root/root         1761 2001-01-23 16:24 vnc/classes/
clipboardFrame.c1ass
-r--r--r-- root/root         3210 2001-01-23 16:24 vnc/classes/
optionsFrame.c1ass
-r--r--r-- root/root         8309 2001-01-23 16:24 vnc/classes/
rfbProto.class
-r--r--r-- root/root         7092 2001-01-23 16:24 vnc/classes/
vncCanvas.c1ass
-r—r—r-- root/root           7100 2001-01-23 16:24 vnc/classes/
vncviewer.c1ass
-r--r--r— root/root         20564 2001-01-23 16:24 vnc/classes/
vncviewer.jar
-r-xr-xr-x root/root        11433 2001-01-23 13:00 vnc/vncpasswd
-r-xr-xr-x root/root        10795 2001-05-17 00:17 vnc/vncserver
-r-xr-xr-x root/root        49685 2001-01-23 13:00 vnc/vncviewer
```

ls -1에서 등록부내용을 목록화한 방법과 비슷하게 파일을 완전한 형태로 목록화하는 방법에 주목을 돌려야 한다.

현행등록부에로 문서고의 내용을 풀기 위해서 c나 t를 x로 바꾼다. 즉

```
$ tar xvf vnc.tar
vnc/
vnc/LICENSE.TXT
vnc/README
vnc/README.vncserver
vnc/Xvnc
vnc/classes/
vnc/classes/DesCipher.c1ass
vnc/classes/ammatedMemoryImageSource. class
vnc/classes/authenticationPanel.c1ass
vnc/classes/clipboardFrame.c1ass
```

```
vnc/classes/optionsFrame.class
vnc/classes/rfbProto.class
vnc/classes/vncCanvas.class
vnc/classes/vncviewer.class
vnc/classes/vncviewer.jar
vnc/vncpasswd
vnc/vncserver
vnc/vncviewer
```

gzip

tar가 파일을 문서고로 만드는데서는 쓸모가 있으나 앞에서 본 실례에서와 같이 그것은 어떤 압축도 수행할수 없다. Linux에서 압축은 보통 gzip지령으로 진행한다.

많은 파일을 하나의 압축된 문서고에서 압축하는 Windows의 Zip문서고와는 달리 gzip는 파일들을 문서고에 압축하지 않고 개별적인 파일들을 간단히 압축한다.

가령 test.pdf라고 하는 특별히 큰 파일이 있는데 얼마동안 그것을 리용하지 않고 그것을 디스크공간에 보관하기 위해 압축하려면 gzip지령을 주어야 한다.

```
$ gzip test.pdf
```

이 지령은 파일을 압축하고 파일이름뒤에 확장자.gz를 붙여 이름을 test.pdf.gz로 바꾼다.

압축하기전에 ls -1지령은 파일크기가 110,778byte라고 현시하였다. 즉

```
-rw-r—r—  1 root   root   110778  Jun 5 16:54 test.pdf
```

압축후에 크기는 83,729byte로 줄어 들었다. 즉

```
-rw-r—r—  1 root   root    83729  Jun 5 16:54 test.pdf.gz
```

이 장에서 서술되는 거의 모든 지령들과 마찬가지로 하나이상의 파일을 목록화하기 위해 임의의 정확한 쉘파일표현식을 쓸수 있다. 실례로

```
$ gzip *
```

지령은 현행등록부(그러나 그것은 보조등록부에는 없다.)의 모든 파일들을 압축한다.

gzip파일의 풀기 gzip파일을 압출하지 않으려면 선택항목 –d가 있는 gzip지령을 리용할수 있다. 즉

```
$ gzip -d test.pdf.gz
```

이 지령은 파일을 압축하지 않으며 확장자 .gz를 삭제한다. 그리고 파일을 test.pdf라는 이름을 가진 초기의 비압축상태로 복귀시킨다.

대리지령인 gunzip에는 선택항목 –d가 없다. 즉

```
$ gunzip test.pdf.gz
```

gzip와 tar의 결합 이전판본에서는 tar지령이 문서고압축을 하지 못하였기때문에 다음실례에서 보여 준것과 같이 gzip를 쓰는것이 보편적이였다.

```
$  tar cvf text.tar *.txt
```

ab.txt
pop.txt
$ **gzip text.tar**

이것은 text.tar.gz라고 하는 압축된 문서고를 만든다.
이 문서고의 내용을 호출하려면 문서고를 풀고 tar를 리용해야 한다. 즉

$ **gunzip text. tar. gz**
$ **tar tvf text.tar**
-rw-r--r-- root/root 48 2001-06-05 16-13 ab.txt
-rw-r—r- root/root 6 2001-06-05 16:13 pop txt

현재의 모든 배포물들에 제공된 tar지령이 들어 있는 최신판본들은 gzip-로 압축된 tar문서고를 직접 호출하거나 만들수 있는 방법을 제공한다.

이전에 본 tar지령에 선택항목 z를 간단히 추가하면 두번째 지령에 필요 없는 압축된 문서고를 만들수 있다. 실례로

$ **tar czvr vnc.tar.gz vnc**
vnc/
vnc/LICENSE.TXT
vnc/README
vnc/README.vncserver
vnc/Xvnc
vnc/classes/
vnc/classes/DesCipher.c1ass
vnc/classes/animatedMemoryImageSource.c1ass
vnc/classes/authenticationPanel.c1ass
vnc/classes/dipboardFrame.c1ass
vnc/classes/optionsFrame.class
vnc/classes/rfbProto.c1ass
vnc/classes/vncCanvas.c1ass
vnc/ct asses/vncviewer.c1ass
vnc/classes/vncviewer.jar
vnc/vncpasswd
vnc/vncserver
vnc/vncviewer

지령은 vnc.tar문서고의 압축된 판본을 만든다. 즉

$ **tar tzvf text.tar.gz**
-rw-r--r-- root/root 48 2001-06-05 16:13 ab.txt
-rw-r---r-- root/root 6 2001-06-05 16:13 pop.txt

또한 text.tar.gz문서고의 내용을 현시하고

$ **tar xzvf text.tar.gz**

ab.txt
pop.txt
지령은 문서고의 내용을 푼다.

주의	tar와 gzip지령에 대한 구체적인 내용과 그 선택항목에 대해서는 이 지령들에 대한 사용설명서페지에서 볼수 있다.

man 및 xman

이제부터 사용설명서에 대한 참고서를 보게 된다. 대부분의 Linux쏘프트웨어에는 표준양식으로 제공되는 사용설명서페지들이 있다. 여러가지 Linux배포물에서 나오는 거의 모든 지령들은 사용설명서페지에서 볼수 있다.

가장 기초적인 형태의 man지령을 주면 현재 있는 임의의 사용설명서페지를 읽을수 있다. 즉

```
$ man command-name
```

이것은 지정된 지령에 대한 사용설명서페지를 현시하며 본문을 현시하기 위하여 less지령을 리용할 때와 같은 방법으로 그것을 탐색도 하고 흐르게 할수도 있다.

지정된 사용설명서페지를 찾을수 없을 때에는 다음과 같은 오유가 나온다.

```
$ man non-existent-man-page
Nomanualentryfornon-existent-man-page
```

이미 X Windows를 잘 알고 있으므로 사용설명서페지를 보기 위하여 man지령과 xman프로그람가운데서 하나를 선택하여 리용할수 있다.

xman을 기동시키자면 다음의 지령을 리용해야 한다.

```
$ xman &
```

또는

```
$ /usr/X11R6/bin/xman &
```

이 지령은 그림 13-1에서와 같은 xman초기창문을 현시한다.

그림 13-1. xman 초기창문

이 초기창문에서 Manual Page단추를 누르면 그림 13-2에서와 같은 xman기본창문이 현시된다.

이 창문의 기본구역은 현재 선택된사용설명서페지의 본문을 현시하는데 리용된다. 처음에 xman의 도움말파일이 현시된다. 또한 창문에는 Options과 Section이라는 두개의 차림표들이 있다.

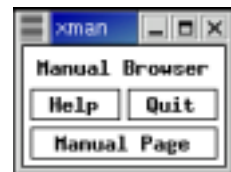



그림 13-3에 보여 준 선택항목차림표를 리용하여 사용설명서페지의 목록(Display Directory)과 현재사용설명서페지(Display Manual Page)사이를 절환할수 있다. 이외에도 현재 사용설명서페지의 내용을 조사한다. Help선택항목은 xman에 대한 완전한 도움말을 현시한다. 여기에는 xman에 대한 구체적인사용법과 같은 지시사항이 상세히 서술

되여 있다. 만일 전문적인 사용설명서페지를 다시 보려면 search선택항목을 선택하고 화면에 있는 본문칸에 지령이름을 입력하면 된다.

그림 13-2. xman기본창문

그림 13-4에 보여 준 section차림표은 읽으려는 사용설명서페지를 찾는 공정들을 포괄하고 있기때문에 아마 더 흥미 있다.

Options
Display Directory
Display Manual Page
Help
Search
Show Both Screens
Remove This Manpage
Open New Manpage
Show Version
Quit

그림 13-3. Option 차림표

그림 13-4. Section 차림표

보는바와 같이 사용설명서페지는 사용자지령(User Commands)과 체계호출(System Calls)을 포함하여 8개의 기본항목들로 나누어 져 있다. 그것들가운데서 하나를 선택하면 그림 13-5에 보여 준것과 같은 등록부목록이 나온다.

임의의 지령이름을 마우스로 두번찰칵하면 그 지령에 대한 사용설명서페지를 읽을 수 있다.

그림 13-5. 등록부목록

▶미리 알아두기

이 장에서는 지령행의 세계에 들어 섬으로써 실제적인 Linux사용자가 되기 위한 첫 걸음을 내디디였다.

지금까지는 X Windows와 그의 기능에 주목하였지만 Unix계렬조작체계의 실질적인 심장부는 지령행에 있다.

제14장 《파일다루기》에서는 Linux가 제공하는 파일 및 등록부로 작업하는데 쓸모있는 지령을 고찰한다. 이 장에서 배운 ls와 같은 지령에 기초하여 등록부와 기타 선진적인 과제에 대해서 연구하게 된다.

제14장. 파일다루기

앞의 장에서는 대략적으로 Linux리용에 무엇이 포함되는가를 보았다. 모든 지령들이 개별적인 파일에 기억된 프로그람을 실제로 어떻게 분리시키는가를 보았다. 특히 파일목록을 표시하는데 쓸수 있는 ls지령과 같은 여러가지 방법에 대하여 고찰함으로써 파일을 다루는 방법에 대한 개념을 가지게 되였다.

이 장에서는 복사, 삭제, 이동, 이름바꾸기 및 만들기 등을 비롯하여 파일과 등록부를 조작하는데 쓰이는 지령들에 대하여 자세히 보게 된다. 이 장에서는 파일이름확장자 즉 DOS와 Windows에서 일반적으로 쓰이는 별표와 같은 통용문자에 대해서 간단히 고찰한다.

파일의 복사와 삭제

대체로 가장 일반적인 두개의 파일다루기과제처리들은 파일의 복사와 이동이다. 여벌복사목적에 따라 가장 값 눅은 망구성방법 즉 다리들을 리용하는《transmit》파일이 아니라 하드디스크구동기에 베껴 놓은 모든 파일이나 플로피디스크에 있는 복사파일을 정상적으로 찾을수 있다. 마찬가지로 일반PC사용자는 부족되는 디스크공간에서 프로그람을 실행하고 값 비싼 디스크공간을 개방하기 위하여 삭제할 파일을 탐색하여 그것을 삭제함으로써 새로운 하드구동기를 구입하지 않아도 된다.

파일의 복사

DOS나 임의의 판본의 Windows에서 DOS재촉문을 조금이라도 써본 사람이라면 대체로 마우스찰칵, 마우스끌기 및 마우스놓기가 파일을 복사하기 위한 방법만이 아니라는것을 알고 있다. 사실 DOS의 copy지령은 통용문자와 같은 보충적인 특성 즉 파일관리기나 Windows Explorer보다 더 빠르고 쉽고 더 강력하게 리용할수 있는 능력을 제공한다.

마찬가지로 Linux세계에서도 cp지령(보통 /bin/cp에서 찾는다.)은 복사를 위해서 리용되며 복사조작을 위한 강력한 도구를 제공한다.

기본복사

cp지령의 가장 기본적인 용도는 같은 등록부에 있는 파일의 사본을 만들기 위해 한 장소로부터 다른 장소에 파일을 복사하는것이다. 가령 현행등록부의 파일(ThisFile)을 같은 등록부의 두번째 파일(ThisFile-Acopy라고 하는)로 복사하려면 다음과 같은 지령을 입력해야 한다.

```
$ cp ThisFile ThisFile-Acopy
```

파일의 등록부목록을 조사하려고 ls -1지령을 주면 날자만 다르고 크기가 같은 두 파일을 찾게 된다. 새 파일에는 복사조작이 수행된 시간을 가리키는 날자가 있다. 이것은 새롭게 구별되는 파일이다. ThisFile-Acopy에로의 변환은 본래의 ThisFile파일에 영향을 주지 않는다.

마찬가지로 /tmp등록부(대체로 다른 사용자와 함께 파일을 공유하기 위해서)의 ThisFile를 복사하려면 다음지령을 리용해야 한다.

```
$ cp ThisFile /tmp
```

그리고 ThisFile을 /tmp에 복사하되 다른 이름을 새 파일에 주려면 다음과 같이 입력해야 한다.

```
$ cp ThisFile /tmp/NewFileName
```

해당 파일에 덧쓰기를 하지 않는다.

Linux배포물에서 주의해야 할 점은 이미 있는 파일우에 새 파일을 복사하여 덧쓰기할 때 실수하기가 쉬운것이다. ThisFile과 NewFile이 둘다 존재하는 경우에는 다음과 같은 지령이 현시된다

```
$ cp ThisFile NewFile
```

이 지령을 주면 NewFile의 내용은 ThisFile의 복사로 덧쓰기되며 이미것은 없어진다(체계의 안전보관에 대해 주의하지 않으면).
이런 문제점을 피하기 위해 복사를 진행할 때 체계가 덧쓰려는 파일을 확인하도록 하는 cp지령의 –i기발을 리용할수 있다. 그러면 복사하려고 할 때 다음과 같은 지령재촉문이 나타난다.

```
$ cp –i ThisFile NewFile
cp: overwrite 'ThisFile'?
```

자기자신을 보호하려면 cp지령에 대한 별명을 만들어야 한다. 지령

```
$ alias cp='cp –i'
```

을 내보내면 임의의 순간에 cp지령을 내보내도록 별명을 정의하며 실제로는 cp–i지령을 발행한다. 이런 방법으로 복사하려는 파일을 덧쓰기하기전에 언제나 지령재촉문이 나타나게 한다. 제16장 《쉘에 대한 리해》의 뒤부분에서 배우게 되겠지만 가입한 시간에 이 별명이 설정되였다는것을 담보하는 .bashrc파일을 리용하여 Bash쉘을 구성할수 있다(쉘은 작업하는 지령행환경을 조종한다. 그러나 그것에 대해서는 후에 보기로 한다.).
다행히도 상급사용자나 뿌리사용자로 가입하는 경우에 이 별명은 Red Hat Linux 7.1에서 기정으로 설정된다. 이것은 뿌리사용자가 체계전반에 대하여 심중한 영향을 미치게 할수 있는것과 같은 약간한 실수도 있을수 있기때문에 매우 중요한 문제이다. 이 기정을 쓰면 위험한 사고가 일어 나지 않게 할수 있다.

한개 지령에 의한 다중파일복사

DOS의 copy지령의 부족점의 하나는 한번에 한개 파일이나 파일표현식밖에 복사할수 없는것이다. 가령 다음과 같은 지령을 주면 /tmp에 한개 파일을 복사한다.

```
$ copy file /tmp
```

또한 다음지령을 입력하면 현행등록부의 모든 본문파일을 /tmp에 복사한다.

```
$ copy *.txt /tmp
```

그러나 복사하려는 3개의 개별적인 파일이 있다면 3개의 지령을 주어야 한다. 마찬가지로 현행등록부의 모든 본문파일과 모든 실행할수 있는 파일을 복사하려면 두개의 지령이 있어야 한다.

Linux의 cp지령은 이것을 좀 더 쉽게 처리하게 한다. cp지령은 DOS판지령에 있는 두 인수보다 더 많은 인수를 가질수 있다. 그러면서도 그 지령에 대하여 여러개의 인수를 통과시킬수 있는데 마지막인수는 도착지로 인식되며 모든 선행하는 파일은 도착지에 복사된다.

실례를 들어 보자. 현행등록부의 파일인 FileOne, FileTwo 및 FileThree를 /tmp에 복사한다고 하자. 3개의 지령을 내보내야 한다는것이 명백하다. 즉

```
$ cp FileOne /tmp
$ cp FileTwo /tmp
$ cp FileThree /tmp
```

그러나 이것들을 하나의 지령으로 모두 한데 묶어서 처리를 쉽게 할수 있다. 즉

$ cp FileOne FileTwo FileThree /tmp

마찬가지로 한개 지령으로 많은 파일을 복사하고 혼합하기 위해 통용문자를 추가할 수 있다. 실례로 지령

$ cp *.txt *.doc *.bak /tmp

은 한 지령으로 3개의 확장자가운데서 임의의 모든 파일을 복사한다.

주의 이 방법으로 다중파일을 복사하면 두개이상의 파일은 하나의 파일로 복사할 수 있기때문에 마지막인수가 등록부로 되여야 한다는것을 명심하는것이 중요하다. 만일 마지막인수를 등록부로 만드는것을 잊어 버리면 다음과 같은 오유통보를 받는다.
cp:when copying multiple files, last argument must be a directory. Try 'cp—help' for more information.

만일 전체 등록부와 그의 모든 보조등록부를 복사하려면 cp지령의 –R기발을 리용할 수 있다. 이 지령은 등록부를 반복하여 복사한다는것을 가리킨다. 만일 SomeDir라고 하는 보조등록부가 현행등록부에 있고 그안에 있는 SomeDir전체를 /tmp의 한 보조등록부에 완전히 복사하려면 다음의 지령을 주어야 한다

$ cp -R SomeDir /tmp

그러면 현행등록부의 SomeDir보조등록부와 같은 /tmp/SomeDir등록부를 생성한다.

개선된 복사

cp지령은 파일과 등록부들에 대한 간단한 복사를 릉가하여 확장되고 개량된 몇가지 특성을 제공한다. 이런 기능들에는 복사에서 파일의 초기상태를 유지하기와 복사과정에 현존파일을 보호하는 대용방법들이 포함된다.

될수록 본래것(original)에 대한 페쇄방법으로 복사물을 만들기 만든 복사물들을 찾는 페쇄방법을 취한다. 복사된 파일의 어떤 특성은 초기파일과 약간 류사한 점을 제공한다는것을 명심해야 한다. 이 특성은 소유권, 허가, 날자 및 기호련결을 포괄한다. 이것들에 대해 하나씩 보기로 하자.

파일을 복사할 때 결과파일은 보통 초기파일의 작성자와 반대로 복사자가 소유한다. user1이 The File이라고 하는 파일을 만들었고 그 파일을 그것들의 홈등록부에 복사하기 위해 user2의 /tmp에 놓았다고 하자. 파일목록을 조사하면 파일은 다음과 같은것을 확인한다.

-rw-r—r— 1 user1 users 16992 Apr 5 12:10 TheFile

후에 user2는 다음지령으로 파일을 복사한다.

$ cp /tmp/TheFile ~/NewFile

결과 파일은 새로운 소유권을 가진다. 그 소유권은 복사자(user2)의것이다.

-rw-rw-r— 1 user2 users 16992 Apr 5 13:10 NewFile

마찬가지로 파일을 등록부에서 작성할 때 그것은 자기에게 할당된 한조의 기정허가를 가진다. 파일을 복사할 때 복사물은 초기파일의 허가를 보존하지 않고 오히려 목적등

록부에 대한 기정허가조를 가진다. 이 실례에서 초기파일과 새 복사물사이에서 허가의 변화에 주목을 돌려야 한다. 초기파일은 오직 그룹을 읽기만 할수 있지만 복사물은 그룹을 쓰기도 할수 있다. 복사물의 날자에 대해서도 역시 변화가 생기는데 그 날자와 시간은 초기파일의 날자도장이 아니라 복사물이 만들어 질 때 반영된다.

그러나 그것들이 복사될 때의 파일의 초기소유자, 날자 및 허가를 보존하려고 하는 경우도 있다. 례를 들어 뿌리회계가 vault의 보관을 위해서 제거할수 있는 하드디스크에 한조의 파일을 복사하는데 리용된다고 하자. 다른 도구를 필요로 하는 정규적인 테프보관과 달리 cp지령으로 이런 류형의 보관을 진행할수 있다. 그러나 이와 같은 보관이 초기파일과 될수록 밀접하게 일치된다는것이 중요하다. 다행히도 cp지령은 그런 속성을 보유한 –p기발을 제공한다. 앞에서 본 실례에 지령

```
$ cp –p /tmp/TheFile
```

을 리용했다면 결과파일은 초기의것과 아주 밀접히 일치된다. 즉

```
-rw-r— 1 user1 users 16992 Apr 5 12:10 TheFile
```

파일복사에서 제기되는 다른 문제는 기호련결을 조종하는 방법이다. 앞장에서 배운것처럼 기호련결은 다른 장소로부터 파일에로 지시자를 제공한다. 이 방법으로 파일이 동시에 하나이상의 장소에 있다고 말할수 있다. 만일 련결을 호출하려고 하면 Linux는 련결이 지시하는 파일을 정확히 호출한다.

일반적으로 기호련결을 복사할 때 결과파일은 같은 파일에 대한 새 련결을 대신하는 련결로 지시되는 파일의 복사물이다. 실례로 TheFile은 다음의 실례에서와 같이 기호련결

```
lrwxrwxrwx 1 user1 users 16992 Apr 5 12:10 TheFile
OtherFile
-rw-r—r-- 1 user1 users 1 Apr 5 11:10 OtherFile
```

을 가진다면 다음의 cp지령을 내보낸다.

```
$ cp /tmp/TheFile ~/NewFile
```

OtherFile의 복사물인 파일에서 결과가 얻어 진다. 즉

```
-rw-rw-r— 1 user2 users 16992 Apr 5 13:10 NewFile
```

그러나 파일대신에 련결을 복사하려고 한다면 무슨 현상이 일어 나겠는가? 결과가 다음과 같이 나타나기를 바란다면 무슨 현상이 일어 나겠는가?

```
lrwxrwxrwx 1 user2 users 2 Apr 5 13:10 NewFile
/tmp/OtherFile
```

마침 CP지령은 장소를 주소화하는 기발을 다시 한번 가진다. 즉 –d기발은 초기파일에 대한 련결을 준다. 간단히 다음과 같은 지령을 주면 요구하는 결과를 얻는다.

```
$ cp –d /tmp/TheFile ~/NewFile
```

이 모든것에 대하여 이제는 결론을 내릴 때가 되였다. 현행등록부와 그의 모든 보조등록부의 쓸모 있는 여벌복사물을 작성하기 위하여 cp지령을 리용하려면 무엇을 해야 하는가? 이 두가지 기발의 조합과 반복적인 복사를 리용하여 목적을 달성할수 있다. 실례로 지령

$ cp –pdR TheDirectory /backups

은 등록부 /backups/The Directory에 있는 TheDirectory의 정확한 복사물을 작성한다. 그러나 cp지령은 이것을 보관하기 위한 간단화된 방법을 제공한다. 즉 –a기발은 하나의 등록부에 대한 문서고가 있어야 한다는것을 가리킨다. 그것은 3개의 기발 –pdR를 지적하는 빠른 방법이다. 즉

$ cp –a TheDirectory /backups

오유예방 앞에서 이미 본바와 같이 오유예방의 한가지 방법은 -i기발을 쓰는것인데 이것은 파일이나 등록부의 복사과정에 덧쓰기가 진행되기전에 호상작용하는 입력재촉을 요구한다. 다른 방법들은 각이한 등급의 보호를 제공하는데 리용할수 있다.

한가지 방법은 cp지령이 덧쓰기되는 어떤 파일의 여벌복사물을 작성하도록 하며 –b 또는 --backup기발을 리용하는것이다. 기정으로 여벌복사물은 초기의 파일이름뒤에 웃물결표식(~)을 가지게 된다. 따라서 다음의 지령을 리용하여 FileOne을 FileTwo에 복사하고

$ cp –b FileOne FileTwo

FileTwo가 이미 존재하고 있다면 여벌복사물은 FileTwo~이라고 하는 초기FileTwo로부터 만들어 진다.

cp지령은 –S기발로 보관파일의 이름을 짓는 방법을 변경시킬수 있다. -S기발은 보관이름에 쓰이는 기정웃물결문자를 어떤 다른것으로 바꿀수 있게 한다. 실례로

$ cp –b –S_ FileOne FileTwo

라고 주면 보관파일이름이 FileTwo__인 결과가 얻어 진다.

선택적으로 --backup기발로 도식들을 이름 짓는 3가지 형태의 보관가운데서 하나를 정의할수 있다.

t 혹은 numbered 차례로 번호가 붙은 보관물을 만든다. 번호가 붙은 보관물이 이미 있다면 이미 있는 보관파일다음의 번호를 차례로 붙인 새로운 보관파일을 만든다. 결국 파일이름은 FileTwo~번호~(실례로 FileTwo~2~ 또는 FileTwo~1~)와 같은 형식으로 나타난다.

nil 혹은 existing 번호가 붙은 보관파일이 이미 존재한다면 그 보관파일을 만든다. 그렇지 않으면 정규적인 단순보관파일을 만든다.

never 또는 simple 기정물결기호 또는 –S기발이 지적하는 선택문자를 리용하여 단순보관파일을 만든다. 이것은 이미 있던 임의의 보관물을 같은 이름으로 덧쓰기한다.

우에서 본 실례에서 번호가 붙은 보관파일을 만들자면 다음과 같은 지령을 주어야 한다.

$ cp backup=t FileOne FileTwo

또는

$ cp –backup=numbered FileOne FileTwo

마찬가지로 다음의 두 지령들도 단순보관파일을 만든다.

$ cp –backup=never FileOne FileTwo
$ cp –backup=simple FileOne FileTwo

–b스위치는 이러한 선택항목을 가진 Red Hat Linux 7.1에서는 동작하지 않는다.

주의	Red Hat Linux 7.1에서는 도식이름을 짓는 보관파일을 조종하는데 아직까지 –V선택항목을 리용한다. 그러나 cp에 대한 앞으로의 공개물에서 이 선택항목을 리용할수 있다고 기대할수는 없다.

선택적인 기발들

cp지령을 위한 많은 기발들이 이 절에서 취급된다는것을 명심해야 한다. 때때로 이 기발들을 기억하기가 힘들다. 다행히 긴 형태의 기발들을 쉽게 기억할수 있는 형태가 있다.

짧은 형태	대신하는 긴 형태
-I	--interactive
-R	--recursive
-P	--preserve
-d	--no-dereference
-a	--archive
-b	--backup
-s	--suffix

처음에는 긴 형태가 보다 직관적인것으로 볼수 있지만 실제로 정규적인 Linux사용자들은 짧은 형태를 리용하게 된다. 긴 형태는 그것을 자주 리용하는 경우에 아주 많은 입력을 요구한다. 실례로 다음과 같은 복합cp지령을 보자.

$ **cp –i -b -V simple –S_ -R ThisDir /tmp**

만일 이 기발들에 대하여 긴 형태를 쓰면 지령은 다음과 같다.

$ **cp –interactive --backup –version-control simple --suffix_--recursive ThisDir /tmp**

얼핏 보기에는 이 지령이 더 읽기가 쉽다는것이 명백하다. 그러나 실제로 등록부를 복사하는데 이것을 간단한 형태로 입력하고 있지 않는가?

파일삭제

파일복사와 마찬가지로 Linux는 /bin/rm에서 본 rm과 같이 파일을 삭제하는 강력한 지령들을 제공한다.

가장 간단한 형태로 rm은 현행등록부에서 하나이상의 파일을 삭제하게 한다. 지령

$ **rm ThisFile**

은 현행등록부에 있는 ThisFile파일을 삭제한다. 그리고 지령

$ **rm *.txt**

는 현행등록부에서 확장자 .txt를 가지는 모든 파일을 삭제한다.

복사에서와 같이 rm지령에 여러개의 인수를 제공할수 있으며 문의된 모든 파일이 삭제된다. 실례로

$ **rm ThisFile *.txt**

는 이미 본 존재하는 두 지령을 결합한것과 같은 동작을 수행한다.

cp지령과 같이 이것은 아주 쓸모 있으면서도 위험하다. 결국 보관문서를 삭제하려고 누구든지 우연히 지령

$ rm thesis.doc

대신에 지령

$ rm thesis.bak

를 주었다면 무슨 일이 일어 나겠는가?

이것은 비슷하지 않는 표현으로 들릴지 모르나 악몽과 같은것이다. 이런 현상이 일어 나면 언제든지 불필요한 작업과 걱정거리가 생긴다.

이런 리유로 해서 rm지령에서는 역시 –i기발을 쓰는것이 좋다. 왜냐하면 그것은 심한 오유를 피하게 하기때문이다.

```
$ rm –i thesis.doc
rm: remove 'thesis.doc'?
```

또한 기정동작을 만드는 rm에 대한 별명을 만들수 있다.

$ alias rm='rm–i'

전체 등록부삭제

사용자는 가끔 전체 등록부를 삭제하려고 한다. 가령 인터네트로부터 내리적재된 쏘프트웨어문서고의 압축자료를 푼 다음 작성된 등록부를 보자. 쏘프트웨어의 설치와 검사를 완전히 끝낸 다음에는 아마도 전체 등록부를 삭제하려 할것이다.

이 작업은 –r기발을 리용해서 수행할수 있다. 실례로 TempInstall이라고 하는 등록부를 삭제하려면 다음과 같은 지령을 주어야 한다.

$ rm -r TempInstall

물론 이 장의 뒤부분에서 보게 되겠지만 매개 삭제를 재촉하는 rm지령에 대한 별명들을 가진다. 이것으로 해서 큰 등록부에 대단히 여유가 많아 진것으로 될수 있다.

```
$ rm -r TempInstall
rm:descend directory 'tempInstall'? y
rm:remove 'TempInstall/File1'? y
rm:remove 'TempInstall/File2'? y

rm:remove directory 'TempInstall'? y
```

수백개의 파일이 있다고 하자. 매개 파일의 입력재촉문에 대답을 주는것은 비현실적이다. 이런 경우에 전체 등록부를 삭제하는것이 완전히 명백할 때에는 rm지령의 –f기발을 써야 한다. 이 기발은 다음의 별명들에 –i기발을 가진 선택적인 조작이 이미 지시되였을 때도 삭제를 요구한다.

$ rm –rf TempInstall

조언

–f기발을 쓰는 경우에 주의해야 한다. 이것은 쓸모 있으면서도 극히 위험하다
Red Hat Linux 7.1에서 상급사용자의 등록자리는 **rm**지령을 위한 기정별명이 **rm**

–i가 되도록 구성된다. 이것은 보기에는 작은 오유라 해도 위험하기때문에 매우 중요하다. 별명을 리용하지 않았을 때 /tmp등록부를 삭제하려는 경우에 생기는 문제를 고려하여 / 와 tmp사이에 공백을 만든다. 즉

```
$ rm –r / tmp
```

이것은 정확히 디스크에 있는 모든 파일과 등록부를 삭제한다. 왜냐하면 rm –i에 대한 별명이 기본이기때문이다.
마찬가지로 상급사용자로 작업할 때 –f기발의 리용에 큰 주의를 돌려야 하며 기본 과제에 대해서만은 상급사용자회계의 리용을 반드시 제한할것을 강조한다. 결국 지령

```
$ rm –rf / tmp
```

는 알맞는 자리에 정확한 별명을 가지지 않으면 아주 위험한것으로 될수 있다.

파일의 옮기기와 이름바꾸기

파일의 옮기기와 파일의 이름바꾸기는 밀접히 련관되여 있다. 따라서 그것들은 이 절에서 한꺼번에 보게 된다. 옮기기와 이름바꾸기가 완전히 다른 작용으로 되는 DOS/Windows 와 Macintosh의 세계에서와는 달리 Linux환경에서는 파일의 이름바꾸기가 파일옮기기의 특수한 경우로 된다.

기본옮기기조작

기본옮기기조작을 고찰하는것으로부터 시작하자. 즉

```
$ mv FileOne /tmp
```

이 지령은 현행등록부의 FileOne파일을 /tmp등록부로 옮긴다.

옮기기와 이름바꾸기

마찬가지로 다음의 지령을 리용하여 /tmp등록부로 파일을 옮기고 파일이름을 바꿀수 있다.

```
$ mv FileOne /tmp/NewFileName
```

이 개념을 리용하면 파일이름을 다시 지을수 있다. 이미 있는 이름으로부터 같은 등록부에 있는 새 이름에로 간단히 파일을 옮긴다. 즉

```
$ mv FileOne NewFileName
```

옮기기와 이름바꾸기가 하나이며 같다고 보겠는가?

하나이상의 파일옮기기

파일의 복사때와 같이 mv지령이 둘이상의 인수를 인정할수 있고 마지막인수가 옮기기에 대한 목적등록부로 제공될수 있기때문에 여러개의 파일을 한번에 옮길수 있다(복사

에서와 같이 여러개의 파일을 한개 파일로는 옮길수 없다. 마지막인수는 하나의 등록부로 되여야 한다.). 확장자가 .bak, .tmp, .old to /tmp인 현행등록부의 모든 파일을 옮기려는 경우를 고찰하자. 즉

$ mv *.bak *.tmp *.old /tmp

이 간단한 지령은 한 조작으로 모든 지정된 파일을 목적등록부(/tmp)로 옮긴다.

어떤 특수한 기발들을 리용하지 않고도 mv지령으로 전체 등록부를 옮길수 있다. 현행등록부에 TheDir이라는 보조등록부가 있고 /tmp의 한 보조등록부가 되도록 그것을 옮기려면 파일에 대하여 이미전에 진행한것과 같이 mv지령을 리용해야 한다. 즉

$ mv TheDir / / tmp

마찬가지로 TheDir를 NewDir라고 하는 등록부의 보조등록부로 만들려면 다음과 같은 지령을 줄수 있다.

$ mv TheDir / /NewDir

주의	cp및 rm에서와 같이 옮기기할 때 실수하여 덧쓰기를 하지 않게 하기 위하여 mv에 대한 별명을 mv –i로 설정하는것이 좋다. $ alias mv= 'mv –i '로 이렇게 할수 있다. Red Hat Linux 7.1에서 이것은 상급사용자등록자리의 cp, rm및 mv지령에 대하여 기정으로 설정된다.

파일만들기

이미 몇가지 파일작성방법에 대하여 보았다. 파일은 복사할 때도 작성되여야 하며 옮기기할 때도 작성되여야 한다. 파일을 작성하는 명백한 기타 방법에는 문서작성, 전자우편부속품의 보관 혹은 화면인입 등이 포함된다.

그러나 파일이 비여 있고 파일의 길이가 령이라도 새 파일을 작성해야 하는 경우가 있다. 이에 대한 가장 명백한 실례는 특수한 상태를 지시하는 파일을 작성하기 위하여 한개 문자가 필요한 경우들이다. 이에 대한 가장 좋은 실례는 file locking이라고 하는 프로그람작성기술이다. 한개 문자가 변경을 만드는 파일을 열면 그것은 역시 다른 프로그람을 지시하는 lock file이라고 하는 특수파일을 작성하며 현행파일은 편집하기 위해 열리며 그래서 변경이 필요 없다는것을 제시한다. 문자가 파일을 일단 닫으면 그 파일을 다시 쓸수 있도록 자물쇠파일을 삭제한다.

이 자물쇠파일을 재빨리 그리고 지내 많은 디스크공간를 요구하지 않고 효률적으로 작성할수 있게 하기 위하여 지정된 이름의 빈 파일을 표시하는 보다 간단한 방법이 필요하다. 이 지령은 파일을 작성한다. 례를 들어 지령

$ touch NewFile

은 이름이 NewFile이고 길이가 령인 새 파일을 작성한다. 즉

-rw-rw-r-- 1 armand armand 0 Apr 6 21:06 NewFile

일반적으로 touch지령은 현존파일의 갱신날자도장을 바꾸는데 리용된다. 많은 프로그람은 그것들이 작업하고 있는 파일의 날자도장에 기초해서 무슨 동작이 일어 나는가를 결정한다. touch지령으로 파일을 열거나 편집하지 않고도 파일의 갱신날자를 바꿀수 있다.

등록부작성

Linux에서 특수한 파일의 한가지 형태는 등록부이다. 등록부는 바로 다른 파일을 포함하는 특수한 파일이다. 다른 사용자가 그 등록부에 무엇이 있는가를 보는것을 허용하거나 금지하는 등록부에 대한 허가를 설정할수 있다. 등록부들은 웃준위뿌리등록부(/)나 개인홈등록부(즉 /home/mj)일수록 간단하다.

그러나 파일을 작성하는데 리용할수 있는것과 같은 지령으로 등록부를 작성할수 있다. 주요지령은 mkdir와 rmdir이다. 만일 홈등록부에서 문건의 등록부를 작성하려면 다음과 같은 지령을 실행할수 있다.

$ mkdir documents

그러나 이것은 홈등록부에 없으면 작업할수 없다. 물론 다음과 같은 지령으로 그 등록부에 필요한 완전한 경로를 지정할수도 있다.

$ mkdir /home/mj/documents

어쨌든 이 지령은 사용자이름과 련관된 특허와 보다 높은 준위의 등록부에 대한 허가를 취급하거나 그것들에 따르지 않을수도 있다. 실례로 정규적인 사용자라면 대체로 새로운 /golf등록부를 작성할수 없지만 뿌리사용자라면 그것을 할수 있다.

물론 등록부를 삭제할수 있다. 례를 들어 다음지령은 이미 작성된 등록부를 삭제한다.

$ rmdir /home/mj/documents

이 지령은 /home/mj 등록부의 허가에 따라 작업하며 이 등록부는 비게 된다. 물론 등록부에 파일이 가득 차 있어도 이 장에서 이미 본 rm -r지령을 리용할수 있다

기호련결프로그람작성

파일작성외에도 기호련결프로그람을 작성하여야 하는 경우도 있다. 기호련결 symbolic links(다른 장소에 있는 실제 파일에 대한 간단한 지시자들인)은 보통 체계관리기들에서 리용한다.개발중에 있는 프로그람에 대한 여러개의 판본을 가지고 있는 프로그람작성자를 보자. 검사하기 위한 현재 판본은 그 개발이 얼마나 추진되였는가에 따라 prog5, prog8 이나 prog10이 될수 있다. 검사과정에 최종판본이 언제든지 실행될수 있다는것을 담보하기 위하여 프로그람작성자는 prog로부터 최종판본까지의 기호련결프로그람을 작성할수 있다. 이렇게 함으로써 prog의 실행은 언제나 응용프로그람이 요구하는 판본을 실행한다.

기호련결프로그람을 작성하는데는 두가지 가능한 방도가 있을수 있다. 첫째는 기호련결프로그람을 지시하는 –s기발을 가진 ln지령을 리용하는것이다. 이 지령은 련결되는 파일과 그 파일에 대한 기호련결프로그람의 장소와 이름을 주는 두개의 인수들을 가지고 있다

례를 들어 현행등록부의 /bin/cp에 대한 련결프로그람 즉MyCopy를 작성하기 위하여 다음의 지령을 리용한다.

$ ln –s /bin/cp MyCopy

MyCopy의 목록을 보기 위하여 ls –l지령을 리용하면 다음과 같은 기호련결프로그람이 나타난다.

lrwxrwxrwx 1 armand armand 7 Apr 6 22:50 MyCopy → /bin/cp

그리 일반적이지 못하지만 기호련결프로그람을 작성하는 다른 방법은 –s기발을 가진 cp지령을 리용하는것이다. 즉

$ cp –s /bin/cp MyCopy

파일이름확장에 대한 안내

이 장을 끝내기전에 중요한 대상 즉 파일이름확장자(*filename expansion*)를 속성으로 조사하기를 보아야 한다. 제16장에서 파일이름확장자에 대하여 Unix쉘을 고찰할때 더 상세하게 설명하겠지만 이 장에서 이미 파일이름확장자의 실례를 보았기때문에 이 절에서는 그것을 문맥으로 취급한다

파일이름확장자는 하나이상의 파일이나 등록부이름들의 목록에 간결한 표현식을 확장하는데 리용되는 특수한 문법이다.이 장에서 본 간단한 실례들은 령 또는 그이상의 문자들을 표시하는 별표 (*)의 리용을 포함한다. 실례로 표현식 *.txt는 다음의 파일이름들가운데서 임의의것과 일치할수 있다.

- .txt
- a.txt
- file.txt
- txt.txt

기정Bash쉘은 파일이름확장자에 대한 풍부한 문법을 제공한다. 이 책의 뒤부분에서 Bash쉘에 대해서 조사하겠지만 두개의 보다 쓸모 있는 기호들을 아래에 소개한다.

?는 임의의 단일문자와 일치하므로 file .?는 file.c일치해야 하지만 file.txt와는 일치하지 말아야 한다.

[문자목록]은 문자목록에서 임의의 단일문자와 일치하므로 file.[abc]는 file.a, file.b , file.c와 일치해야 하지만 file.d나 file.txt와는 일치하지 말아야 한다. 선택적으로 file.[x-z]는 file.x, file.y, file.z와 일치해야 한다.

▶미리 알아두기

우리는 긴 로정을 걸어 왔다. 지금은 X Windows를 리용하는 방법을 알게 되였고 지령행과 쉘과 같은 Linux의 현실적인 심도를 파고들기 시작하였다. 지령-입력재촉환경에서 리용할수 있는 몇가지 주요한 프로그람들에 대하여 배웠고 간결하면서도 강력한 지령을 리용하여 세련된 방법으로 파일들을 다룰수 있게 되였다.

제15장 《체계구성을 위한 Linuxconf 와 기타 도구의 리용》을 취급하는 다음단계에서는 Linuxconf를 비롯한 여러가지 Red Hat Linux 도구를 리용하여 일부 중요한 체계구성과 관리과제처리를 수행하는 방법을 배우게 된다. 이 기능들은 Linux에서 리용하는 인쇄기들을 설정하고 사용자를 관리한다.

제15장. 체계구성을 위한 Linuxconf와 기타 도구의 리용

Linux에 아주 익숙된 사용자들과 체계관리기들이 자기식의 기능들을 늘이고 자기들의 체계에 대한 동작을 조종하는 파일을 수동편집하는 작업에 접근하려고 하지만 Linux를 생산적인 도구로 리용하려는 일반콤퓨터사용자들은 이것이 심각하고 감당할수 없는 사업이라는것을 발견하게 된다.

사실상 구성파일을 수동편집할수 있다는 착상은 그것이 Linux가 사용자에게 친절하지 못한것처럼 보이고 Windows가 사용자에게 친절하다고 널리 알려 진 주되는 리유의 하나로 될수 있는 아주 심각한 문제이다.

그러나 그가운데서도 Red Hat는 기초를 이루는 기본Linux워크스테이션의 구성에 대한 처리로부터 사용자를 보호하게 하는 자기들의 Linux배포물에 속한 도구들을 통합시키기 위해서 오래동안 노력해 왔다. 수많은 도형처리구성도구는 사용자, 인쇄기, 시계기구, 모뎀, 망, 쏘프트웨어제품 등을 관리하는데 리용할수 있다.

이 도구들가운데서 가장 포괄적인것이 Linuxconf이다. 이 《all-in-one》편의프로그람은 리용하기 편리한 도형방식대화칸과 창문을 통하여 가장 일반적인 체계관리과제를 수행하기 위한 기능들을 제공한다. Linuxconf는 체계관리를 위한 선택적인 도형방식대면부를 제공한다.

주의	다른 Linux배포물들도 역시 Corel의 조종쎈터, Caldera의 WebMin 그리고 S.u.S.E의 YaST를 포함하는 Linuxconf와 비슷한 기능을 가진 도형방식도구들을 가지고 있다.

이 장에서는 한쌍의 Red Hat Linux 7.1도형방식도구 즉 Printconf편의프로그람과 Linuxconf에서의 숨은 조사를 고찰하고 Linuxconf의 몇가지 특성들의 상세한 적용을 제공한다.

인쇄기의 구성

대체로 Linux에서 아주 시끄러운 구성과제들가운데서 하나는 필요한 체계구성파일을 수동으로 편집하는 인쇄기작업 특히 PCL인쇄기를 얻으려고 하는것이다.

Red Hat는 Printconf편의프로그람을 리용하여 이 과제를 관리할수 있게 하는 작업(간단치 않다면)을 멋들어 지게 수행할수 있다. GNOME의 기본차림표로부터 Program→System→Printer Configuration을 선택하여 이 편의프로그람을 실행할수 있다. 뿌리사용자로 가입하지 않는 뿌리사용자통과암호를 입력하라는 재촉문이 나타난다. 그러면 이것은 그림 15-1에서 보여 준 Red Hat Linux의 인쇄기구성도구를 불러 낸다.

주의	KDE에서는 KDE기본차림표로부터 System→Printer Configuration을 선택하여 이와 같은 도구를 불러 낼수 있다.

아직까지 Printconf는 구성파일인 /etc/printcap를 구성하기 위한 가장 편리한 도구이다. 사실상 Printconf가 있으면 /etc/printcap파일을 직접 편집하기 위한 가정이 더는

필요 없다. 수동으로만 구성할수 있는 인쇄기를 가지고 있다면 자기들의 설정들을 /etc/printcap.local 파일에 추가할수 있다.

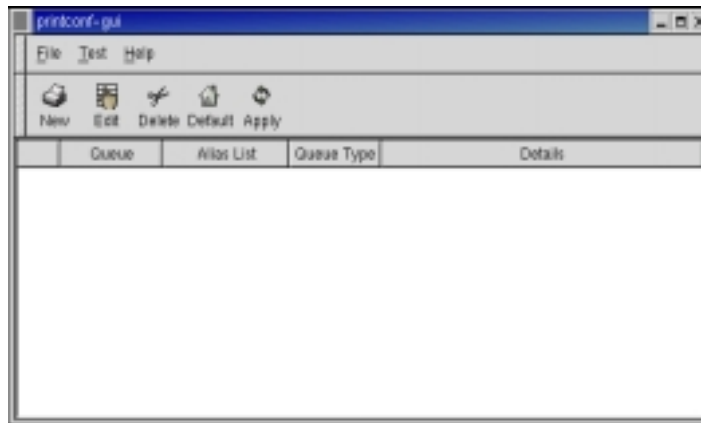



그림 15-1. 인쇄기구성도구

새 인쇄기는 4단계의 처리과정으로 설정한다. 우선 이름을 추가한다. 여기에는 새 인쇄기의 이름을 더 쉽게 기억하게 하는 별명들이 포함될수 있다. 다음대기렬을 설정한다. 대기렬들은 망과 분리된 국부상태로 될수 있다. 설정한 대기렬은 역시 장치와 관련된다. 또한 Linux가 인쇄기와 통신하는것을 방조하는 구동프로그람을 설치한다. 마지막으로 변화를 보관하고 Line Printer 데몬 즉 lpd를 재시동한다.

인쇄기의 추가

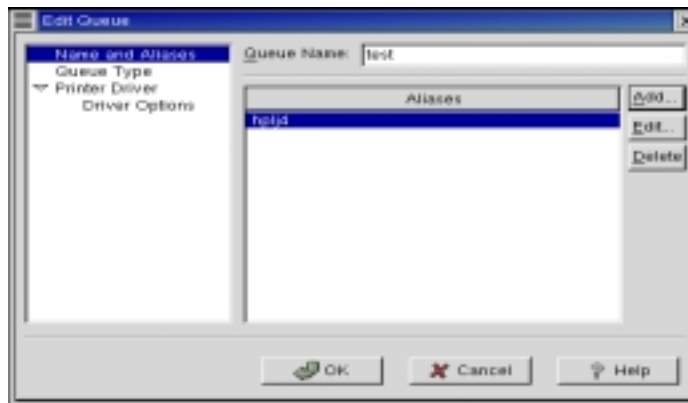



그림 15-2. 이름과 별명의 선택

처리를 시작하기 위하여 Printconf도구띠의 New단추를 찰칵한다. 그러면 그림 15-2에 보여 준 Edit Queue대화칸이 나타난다. Queue Name본문칸에 인쇄기의 이름을 입력한다. 만일 하나이상의 이름이 요구되면 하나이상의 별명들을 설정할수 있다. Add단추를 찰칵하고 선택한 이름을 입력한다. 추가적으로 별명들을 더 요구하면 반복해야 한다.

그림 15-2에서 인쇄기의 이름은 test이며 그것은 hplj4라는 별명을 가진다.

대기렬의 설정

인쇄기들에 대기렬을 설정한다. Edit Queue대화칸에서 Queue Type를 찰칵한다. 그림 15-3에 보여 준것처럼 구성을 위한 두개의 항목 즉 인쇄기와 지정된 인쇄장치의 장소가 있다.

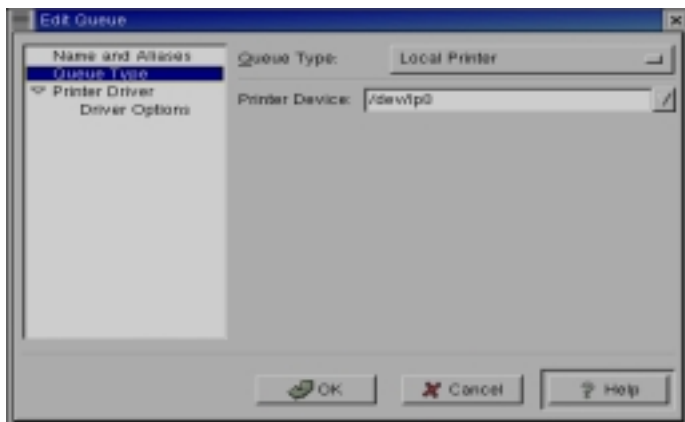



그림 15-3. 대기렬과 장치의 선택

각이한 류형의 5개 대기렬을 리용할수 있다. 가장 간단한것은 Local Printer인데 여기서 인쇄기는 병렬포구나 USB포구를 거쳐 콤퓨터와 직접 련결된다. Local Printer를 찰칵하면 Printconf는 5개의 Queue Type선택항목들을 모두 보여 준다. 다른 4개의 선택항목들은 인쇄기가 다음의 방법들가운데서 어느 한가지로 망과 접속된다는것을 념두에 둔것이다.

Unix Printer(lpd Queue) 목표인쇄기가 망을 통해서 Unix나 다른 Linux콤퓨터에 접속되고 콤퓨터가 그것들과 통신하기 위해 망파일체계(NFS : Network File System) 규약을 리용한다면 이 선택항목을 리용해야 한다. 그 콤퓨터에 대한 대기렬이름은 물론 인쇄봉사기콤퓨터의 이름도 필요하다.

Windows Printer(SMB Share) 목표인쇄기가 망에 있는 Microsoft Windows콤퓨터에 접속되거나 Linux 와(혹은) Unix콤퓨터들이 Samba를 리용하여 망에 접속된다면 그 인쇄기와 접속하는 대기렬의 형을 설정할수 있다. 인쇄기에 지정된 할당이름과 인쇄기가 있는 콤퓨터에 지정된 IP주소가 있어야 한다. 망할당형태에 따라 Windows작업그룹의 이름, 사용자이름 그리고(혹은) 인쇄기를 호출하는데 필요한 통과암호도 역시 요구될수 있다.

일러두기	할당이름은 Microsoft Windows콤퓨터에 련결된 인쇄기에 대한 완전한 경로이다. 례를 들어 Windows콤퓨터이름이 MSWin1이고 인쇄기할당이름이 HPLaserJ라면 리용하는 할당이름은 //MSWin1/HPLaserJ이다.

Novell Printer(NCP Queue) 자기의 망이 통신을 위해서 어떤 Novell 망규약을 리용하면 NCP 대기렬형은 자기를 위한것이다. 이 방식으로 련결된 인쇄기에서는 적용할수 있는 사용자이름과 통과암호는 물론 인쇄봉사기콤퓨터의 이름, 그 콤퓨터에 있는 대기렬

의 이름을 요구한다.

JetDirect Printer JetDirect Printer대기렬형은 망우의 임의의 콤퓨터와의 어떤 직접적인 물리접속이 없이 망에 직접 련결되는 HP인쇄기와 HP호환인쇄기를 위한것이다. 기정값 9100이 아니면 포구번호는 물론 이 인쇄기에 지정된 IP주소를 요구한다.

인쇄기대기렬을 설정한 다음 인쇄기장치를 선택할수 있다. Red Hat Linux 에는 표준인쇄기와 USB인쇄기에 리용할수 있는 인쇄기장치들이 있는데 표준인쇄기에 대해서는 /dev/lpo, /dev/lp1, dev/lp2가 있고 USB인쇄기에 대해서는 /dev/usb/lpo~/dev/usb/lp15가 있다.

인쇄기구동프로그람의 구성

다음단계는 Linux가 목표인쇄기와 통신할수 있는 인쇄기구동프로그람을 구성하는것이다. Edit Queue대화칸에서 Printer Driver를 찰칵하고 각이한 4가지 종류의 인쇄기들로부터 선택한다. 그림 15-4에 그중의 3가지를 보여 주었다.

Postscript Printer 대부분의 경우 포스트스크립트인쇄기를 가지고 있다면 표준포스트스크립트구동프로그람을 동작시킬수 있다.

Text Only Printer(본문전용인쇄기) 이 항목을 선택하면 표준본문파일에서 본문정보만 인쇄하거나 인쇄기가 표준ASCII본문코드만 처리할수 있다.

Raw Print Queue (행인쇄대기렬) 행인쇄대기렬에는 인쇄기구동프로그람이 없다. 이미 인쇄출력을 자기 인쇄기가 리용할수 있는 자료로 변환하는 프로그람을 가지고 있다면 이 선택항목을 실행할수 있다.

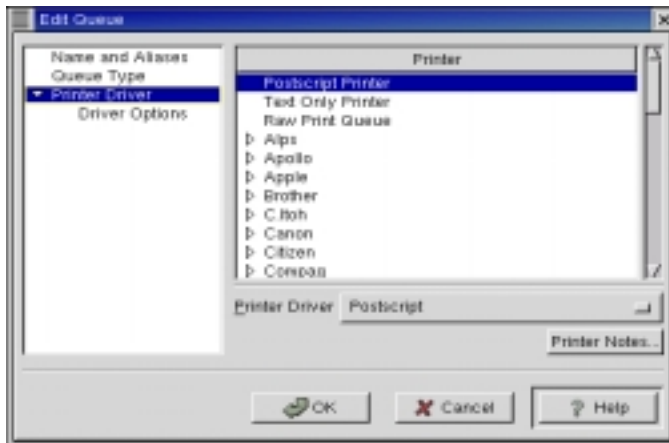



그림 15-4. 인쇄기구동프로그람의 선택

Brand Specific(특수상표) PrintConf 편의프로그람에는 인쇄기형들과 관련된 많은 인쇄기구동프로그람이 있다. 포스트스크립트인쇄기가 없으면 그 인쇄기와 관련된 회사와 자호를 선택한다.

주의 일부 인쇄기는 하나이상의 구동기프로그람과 련관되여 있다. 거의 모든 경우에 기정구동프로그람은 잘 동작한다. 그러나 문제가 제기되면 자기 인쇄기를 위한 대리구동기프로그람의 목록을 제안하기 위해 실제로 있는 인쇄기구동프로그람의 이름을 찰칵한다.

각이한 구동프로그람들에 대해서 같은 인쇄기를 설정할수도 있다. 례를 들어 인쇄자료를 인쇄기가 리용할수 있는것으로 변환하는 특수한 프로그람을 가지고 있으면 구동프로그람이 필요 없으며 행인쇄대기렬을 리용하는 인쇄기를 설정할수 있다. 인쇄요구의 정지를 취급하는 구동프로그람에 대해서도 같은 인쇄기를 설정할수 있다. 매개 기능에서 각이한 이름, 별명 그리고(혹은) 인쇄장치들을 리용하는가를 정확히 확인해야 한다.

일단 인쇄기구동프로그람을 선택하면 각이한 구동기프로그람의 선택항목들로 그것을 주문하여 만들수 있다. 례를 들어 그림 15-5는 HP Laser Jet 4L인쇄기와 련관된 구동기프로그람선택항목들을 주었다.

리용할수 있는 선택항목들에는 많은 변종이 있다. 인쇄기구동프로그람을 전용화하는 가장 일반적인 선택항목들을 아래에 주었다.

Send EOT(전송끝의 보내기) EOT는 《End of Transmission(전송끝)》의 준말이다. 인쇄기가 련속적인 인쇄작업의 시작이나 끝에서 고장나면 이 선택항목을 리용할수 있다.

Rerender Postscript(포스트스크립트의 되돌리기기) 이 선택항목은 자기의 파일을 포스트스크립트파일로 다시 전송한다. 일부 인쇄기는 도형방식을 조종하기 위하여 포스트스크립트양식을 요구한다. 그러나 이 선택항목은 일부 다른 인쇄기가 포스트스크립트자료를 처리할수 없기때문에 작업할수 없다.

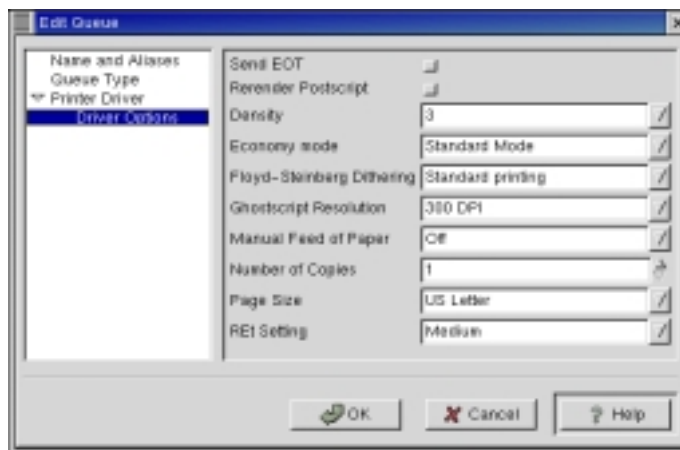



그림 15-5. 인쇄기구동프로그람선택항목

Page Size(페지크기) 이 선택항목은 인쇄기에서 종이크기를 설정한다.

리용할수 있는 다른 선택항목들은 보통 인쇄기의 특성과 관련된것이다. 보다 많은 정보를 위하여 인쇄기의 문서와(혹은) www.linuxprinting.org에서 리용할수 있는 Linux 인쇄문서를 참고한다.

일단 구성작업을 끝냈으면 Edit Queue 대화칸에서 탈퇴하기 위해서 Ok단추를 찰칵한다.

구성검사

검사를 하기전에 /etc/printcap구성파일에서 구성한것을 보관하여야 한다. Printconf의 기본화면에 돌아 가서 이 파일에 변환을 보관하기 위해 File→Save Changes를 선택한다. 그다음 Line Printer 데몬이 이 파일을 다시 읽도록 하기 위해 File→Restart→Lpd를 선택한다.

이제는 구성을 검사할수 있다. Test차림표에는 3개의 선택항목들이 있다.

Print Postscript test Page 표준 Red Hat Postscript 검사페지에는 본문, 색띠 그리고 8.5 X 11in인쇄출구의 모서리로부터 0.5및 1.0in 떨어 진 경계선이 포함되여 있다.

Print A4 Postscript test Page 이것은 유럽에서 보다 일반적인 A4용지를 전용으로 만든 표준 Red Hat Postscript검사페지이다.

Print ASCII Test Page 이것은 도형이 없는 표준본문인쇄출력이다.

현존인쇄기의 관리

일단 인쇄기가 설정되였으면 인쇄기의 구성을 간단히 편집할수 있다. 언급된 인쇄기를 선택하고 도구띠의 Edit를 찰칵한다. 이렇게 하면 앞에서 본것과 같은 대화칸이 나타난다.

Printconf의 선택

Printconf를 위해서 주요하게 두가지 즉 PrintTool 과 Apsfilter를 선택할수 있다. PrintTool은 7.0판본까지의 Red Hat에서 리용된 편의프로그람이다. Apsfilter는 제18장《주변장치의 리용》에서 보게 된다. Red Hat를 써본적이 있으면 PrintTool을 리용하는 것이 더 편리할수 있다. PrintTool을 설정하기 위해서 우선 Pringconf를 해제한다. 뿌리사용자방식에서 이것은 다음과 같은 순서로 된 rpm지령으로 수행할수 있다.

```
# rpm -e printconf-gui
# rpm -e printconf
```

다음 Red Hat Linux 7.0 **CD-ROM**으로부터 또는 www.rpmfind.net와 같은 원천으로부터 그것을 내리적재하여 PrintTool을 설치할수 있다. 이때 다음과 같은 두개의 프로그람묶음이 있어야 한다.

```
rhs-printfilters-1.81-1.i386.rpm
printtool-3.54-1.i386.rpm
```

이 프로그람묶음이 /tmp등록부에 내리적재되였다면 뿌리사용자방식에서 그것들은 다음과 같은 순서로 된 rpm지령으로 설치할수 있다.

```
# rpm -i /tmp/rhs-printfilters-1.81-1.i386.rpm
# rpm -i /tmp/printtool-3.54-1.i386.rpm
```

Linuxconf의 기동

Red Hat Linux 7.1로 기동하면 Linuxconf는 더는 기정으로 설치되지 않는다. 사실 련관된 rpm 프로그람묶음은 이 책과 함께 배포된 **CD-ROM**에 포함된적이 없었다. 그러나 두장이상의 **CD-ROM**들을 가지고 완전한 설치를 했으면 두번째 설치 CD-ROM으로 Linuxconf프로그람묶음인 linuxconf-1.24r2-10.i386.rpm을 설치할수 있다.

구성에 따라 CD-ROM은 CD를 구동기에 끼워 넣은후 실장될수 있다. 이것을 검사하기 위해 **moun**t지령을 입력한다. 결과목록에 다음과 같은 행이 보이면 CD-ROM은 이미 설치되였다.

```
/dev/hdb on /mnt/cdrom type is09660
```

/mnt/cdrom등록부에서 요구하면 CD-ROM을 설치한다. 일단 설치하였으면 다음지령으로 Linuxconf를 설치할수 있다.

```
$ rpm -i /mnt/cdrom/RedHat/RPMS/linuxconf-1.24r2-10.i386.rpm
$ rpm -i /mnt/cdrom/RedHat/RPMS/gnome-linuxconf-0.64-1.i386.rpm
```

주의 Red Hat Linux 7.1의 두번째 설치 CD-ROM이 없으면 이 프로그람들을 ftp.redhat.com이나 www.rpmfind.net와 같은 다른 원천프로그람으로부터 내리적재할수 있다.

일단 설치되면 Linuxconf는 xterm창문의 지령재촉문에 다음의 지령을 입력하여 지령행으로부터 기동할수 있다.

```
$ linuxconf-auth
```

뿌리사용자가 아니면 Red Hat Linux 7.1은 Linuxconf를 기동하기전에 뿌리통과암호를 재촉한다.

Linuxconf가 기동하면 그림 15-6과 같은 창문이 나타난다.

주의 Linuxconf를 기동하는 첫 순간에 역시 환영화면을 보게 된다. 이 화면을 보면 Linuxconf를 시동하기 위해서 Quit를 찰칵한다.

Linuxconf는 망의 구성, 사용자와 그룹의 관리를 포함하는 Linux환경의 많은 Linuxconf는 망구성, 사용자와 그룹관리를 비롯한 각이한 양상들을 조종하는데 리용될수 있다. Linuxconf의 이러한 각이한 항목들은 창문의 중심에 있는 확장할수 있는 나무계층구조로부터 선택될수 있다. 확장할수 있고 접을수 있는 이 나무는 거의나 Windows 98/Me와 NT/2000 Explorer의 파일과 등록부를 조종하는 나무와 같은 방법으로 작업한다.

그림 15-6. Linuxconf 창문

모듈의 추가

Linuxconf를 처음 시동하였을 때 필요한 모든 도구들이 다 있을수는 없다. 다행히도 모듈을 쉽게 추가할수 있다. 리용할수 있는 모듈들을 조사하기 위하여 Control표쪽을 찰칵하고 그림 15-7에서 보여 준 List Of Modules(모듈목록)표쪽을 펼치기 위하여 Control Files And Systems ➡Configure Linuxconf Modules를 선택한다. 이 장의 뒤부분에 맞는 설명을 위해서 적어도 modemconf모뎀구성도구를 선택해야 한다.

이 도구를 기동시키자면 modemconf를 선택하고 List Of Modules표쪽에 있는 Accept단추를 찰칵해야 한다. 그다음 도구띠에서 File➡Act/Change를 선택하고 또 File➡Quit를 선택한다. Linuxconf를 재기동하면 Modem 선택항목이Config표쪽의 Peripherals들에 나타난다.

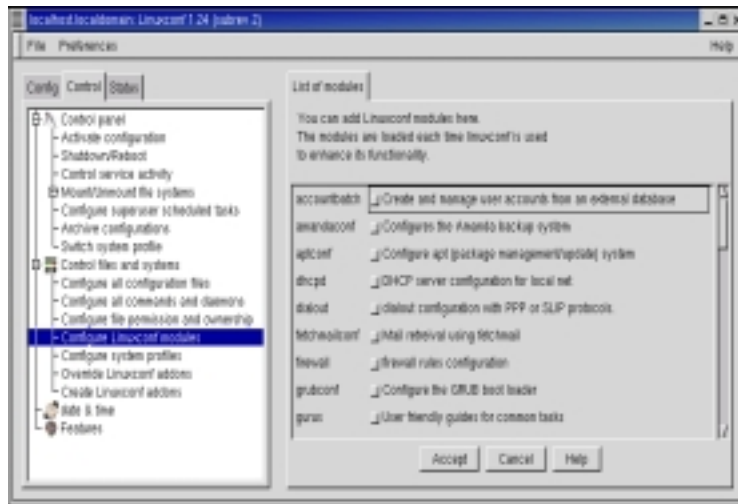



그림 15-7. Linuxconf List Of Modules표쪽

주의

Linuxconf로부터 탈퇴하기 위해서 두개의 지령 즉 File➡Quit와 File➡Quit(no check)를 리용할수 있다. 두번째 선택항목은 Linux체계의 상태와 함께 구성을 구체적으로 검토한다. 차이가 있으면 체계를 갱신하기 위해서 Linuxconf를 제공한다.

사용자 및 그룹관리

이 절에서는 사용자와 그룹을 관리하는 Linuxconf의 리용에 중심을 두고 고찰한다.

사용자와 그룹관리방법은 Linuxconf창문의 왼쪽에서 선택나무의 Users Accounts ➡Normal 항목에 있는 Config표쪽에서 찾을수 있다.

사용자관리

사용자를 관리하자면 Linuxconf의 Config표쪽에 가서 Users Accounts➡Normal➡User Accounts를 선택해야 한다. 그러면 그림 15-8에 보여 준 기정사용자관리화면이 펼쳐 진다. 이 화면은 현재의 사용자에 대하여 상세하게 렬거하며 사용자목록의 아래에는 기능단추렬이 있다.

이 화면을 리용하여 다음과 같은 작업을 할수 있다.

- 새로운 사용자를 추가한다.
- 이미 있는 사용자들에 대한 속성을 편집하거나 자기들의 현재 특성을 조사한다.
- 사용자를 감금하거나 해제한다.
- 사용자들을 체계에서 완전히 삭제한다.

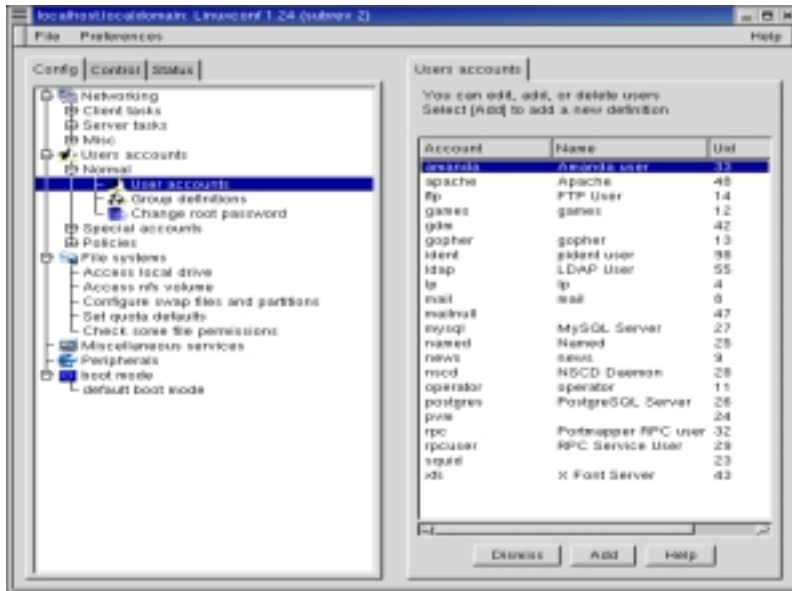



그림 15-8. 사용자목록과 사용자를 관리하는 기능단추렬의 리용

새 사용자의 추가

새 사용자를 추가하기 위하여 Add단추를 찰칵한다. 그러면 그림 15-9에 보여 준 User Account Creation(사용자회계작성)화면이 나타난다. 대화칸의 거의 모든 정보마당은 비여 있으며 새 사용자에 대한 정보를 입력하기를 기다리고 있다. 여기에 개괄한것처럼 정보입력이 끝나면 사용자를 작성하기 위하여 Accept단추를 찰칵한다.

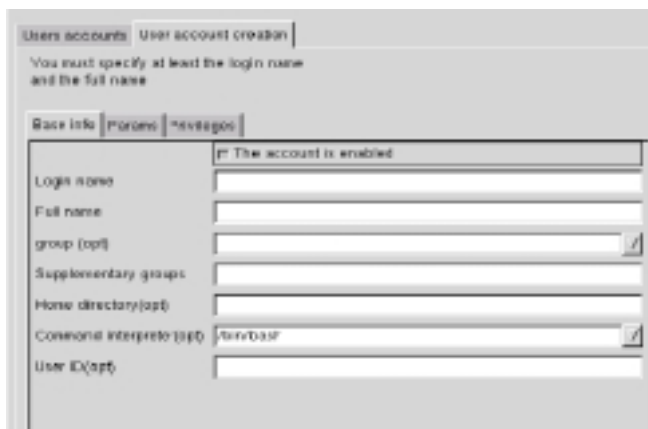



그림 15-9. 새 사용자추가

Login Name (가입이름) 칸에 있는 첫번째 마당은 Login Name(가입이름)마당이다.

이 마당에 사용자이름 즉 체계에 가입하려는 사용자의 이름을 입력한다. 그러나 그것이 완전한 이름은 아니다. 이 사용자이름은 공백이 없어야 하며 문자나 수자로 시작하여야 한다.

실례로 이름이 Arman Danesh인 사용자에 대해서 다음과 같은 사용자이름들가운데 어느 하나를 선택할수 있는데 이 사용자이름들은 모두 선택된 사용자 이름과 이러저러하게 련관되여 있다.

- armand
- adanesh
- arman
- danesh
- arman_danesh
- ArmanDanesh

Full Name (완전한 이름) Full Name마당은 사용자의 완전한 이름을 입력하는 곳이다. 실례로 완전한 이름은 사용자이름으로부터 끌어 낸 사용자의 전자우편주소와 완전한 이름을 둘다 포함하는 사용자의 전자우편에 대한 From(출발)선을 구성하는데 리용된다. 이 마당은 공백을 포함할수 있다.

Group and Supplementary Groups (그룹과 보조그룹) Group마당은 사용자를 지정된 기정그룹에 할당한다. 왼쪽 공백이면 Red Hat Linux는 사용자에게 사용자의 사용자 ID번호에 해당한 그룹 ID번호를 가진 유일한 그룹을 준다. 무엇을 하고 있는지 정확히 모르는 한 가장 좋기는 Linuxconf가 할당한 미리 선택된 값을 리용하지 않는것이 좋을것이다.

Supplement Group마당은 보호와 접근조종목적을 위해 추가그룹에 새 사용자를 할당하는데 리용될수 있다. 대부분의 소규모체계와 망에서는 이것이 필요 없다.

Home Directory(홈등록부) Home Directory 마당은 사용자의 홈등록부에 대한 완전한 경로를 지시한다. Red Hat Linux체계에서 이것은 기정으로 /home등록부의 보조등록부이며 등록부이름은 사용자이름과 같다. 실제로 Login Name마당에 사용자이름을 입력하면 Linuxconf는 /home에 기정홈등록부를 할당하기 위해서 그것을 자동적으로 리용한다. Home Directory마당의 선택등록부경로를 제공하여 이 기정을 취소할수 있다.

례를 들어 사용자이름이 armand인 사용자에 대한 홈등록부는 /home/armand로 될수 있다. 대규모다중사용자봉사기가 없거나 체계우에 홈등록부를 배치하는 반복계획을 실행하지 않는 한 기정항목은 남아 있다.

Command Interpreter(지령해석프로그람) Command Interpreter 마당은 사용자의 지령행환경이 작용하는 통로를 조절하는 사용자의 기정쉘을 지시하는데 리용된다(여러가지 쉘과 그것들의 특성을 배우려면 제16장 《쉘에 대한 리해》를 보아야 한다).

거의 모든 Linux판본들에서 모든 새 사용자들에 대한 기정쉘은 Bash쉘이다. Command Interpreter마당의 옆에 있는 내림차림표를 찰칵하면 Ash쉘, C-쉘, Enhan C-쉘(/bin/tcsh)과 같은 여러가지 종류의 Bash쉘을 포함하는 대리쉘의 목록이 제시된다. 일반적으로 기정쉘을 바꿀 필요가 없다.

Creating the User(사용자작성) 새 사용자에 대한 필요한 정보를 다 제공했을 때는 사용자를 추가하기 위하여 Accept단추를 찰칵한다. 일단 Accept를 찰칵하면 두번에 걸쳐 사용자통과암호에 대해 입력재촉을 받는다(통과암호를 확증하는것이 필요하기때문에 두번에 걸쳐 입력재촉을 받는다.). 통과암호재촉문은 그림 15-10에 있는것과 같이 보인다.

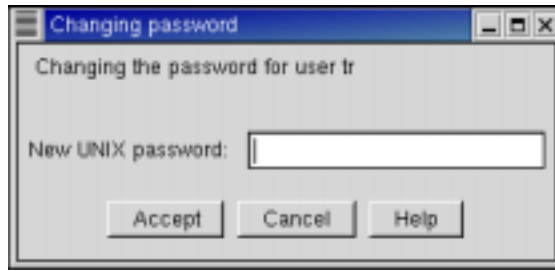



그림 15-10. New UNIX password재촉문

일단 두번에 걸쳐 통과암호를 입력하고 매번 Accept를 찰칵했으면 이미 먼저 그림 15-8에서 본 기본사용자관리화면이 다시 나타나며 다음에는 새 사용자가 현행사용자목록에 나타난다.

주의 통과암호로 사전단어를 리용하면 Linuxconf는 그에 대한 오유통보문을 내보낸다. 그래도 사전단어를 계속 통과암호로 리용하려면 Accept를 찰칵하고 다음 Retype New UNIX Password(새 UNIX통과암호를 재입력)본문칸에서와 같은 통과암호를 입력한다.

일러두기 알맞는 통과암호는 문자와 수자의 조합을 포함한다. 례를 들어 ltr29tmr와 같은 통과암호는 어떤 정규적인 단어보다 파괴하기가 훨씬 더 어렵다. 그런 통과암호는 기억하기 쉬워야 한다. 즉 ltr29tmr는 《I take route 29 to my restaurant》를 의미한다.

사용자의 편집 또는 보기

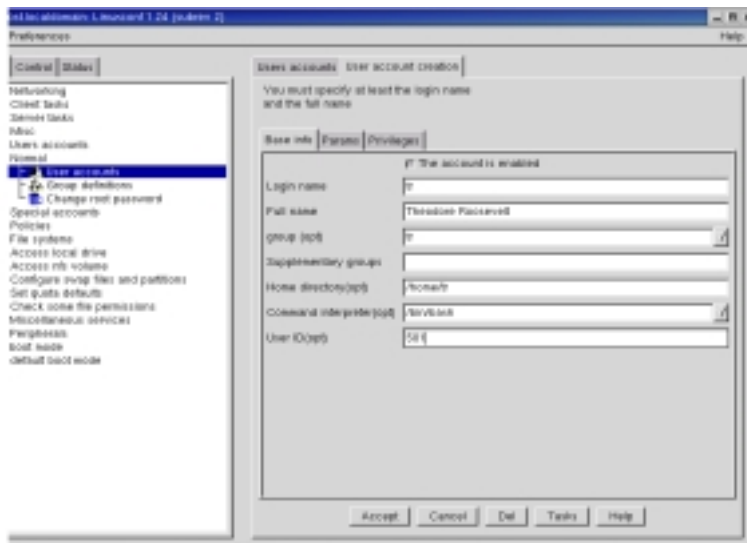



그림 15-11. 사용자정보의 보기 및 편집

사용자에 대한 구체적인 자료를 보거나 정보를 바꾸기 위하여 기본사용자관리화면의 사용자목록에서 이름을 찰칵한다. 화면칸은 그림 15-11에 있는것과 같이 나타난다. 이 화면칸은 한가지 기본적인 차이를 내놓고는 새로운 사용자를 추가하려고 할 때 본것과 같다. 즉 모든 마당은 목록에서 선택된 사용자와 관련되는 정보들을 포함한다. 사용자에 대한 임의의 정보를 바꾸기 위해서 이 형태를 리용할수 있다.

만일 사용자를 바꾸고 이 변경을 사용자목록에 넘기려면 간단히 Accept를 찰칵한다. 한편 변경을 보관하지 않고 대화칸을 벗어 나기 위해서는 Cancel을 찰칵한다.

사용자의 허가와 무시

사용자의 허가와 무시는 쓸모 있는 개념이다. 사용자를 무시한다는 리면에 놓은 의도는 체계로부터 그것을 영원히 삭제하지 않고 림시로 사용자의 회계를 비능동상태로 만드는것이다. 이것은 사용자가 후에 복귀될수 있게 한다. 이런 류형의 기능이 리용되는 경우의 실례들은 사용자가 료금을 지불하지 않고 그들의 접근을 차지했을 때 림시로 접근특허를 금지하는 인터네트봉사제공자(ISP)를 포함해야 한다.

사용자의 회계를 무시하기 위해서 사용자목록에서 사용자를 찰칵하고 The Account Is Enabled의 선택을 취소한다. 변경들을 접수하면 사용자의 회계는 무시된다.

사용자의 허가 무시된 사용자를 허가하기 위하여 간단히 사용자목록에 있는 사용자의 이름을 찰칵하고 그 사용자의 편집화면에서 The Account Is Enabled를 선택한다. Accept를 찰칵하면 변화들을 금지하고 사용자의 회계가 허가된다.

사용자의 무시 사용자관리에서 마지막선택항목은 사용자를 체계로부터 삭제하는것이다. 일단 그렇게 하면 사용자의 회계를 다시 작성함으로써만 사용자를 회복시킬수 있다. 사용자를 삭제하기 위하여 그 사용자에 대한 편집화면을 펼치는 사용자목록의 사용자이름을 찰칵한다. 다음 편집화면의 밑에 있는 Del단추를 찰칵한다. 그러면 그림 15-12에 있는것과 같이 확인화면이 나타난다.

그림 15-12. Deleting Account확인화면

회계삭제확인화면에는 3개의 선택항목들이 있다. 즉

Archive The Account's Data (회계자료의 보관) 사용자의 회계(홈등록부, 할당량 등)로부터 자료는 /home/oldaccounts 등록부에 보관되며 후에 호출하기 위하여 압축된다.

Delete The Accont's Data (회계자료의 삭제) 사용자회계로부터 자료가 완전히 삭제된다.

Leave The Accont's Data In Place(알맞는 장소에 회계자료를 남기기)사용자회계

가 더는 존재하지 않을 때에도 사용자의 회계로부터 자료는 남아 있게 된다.
사용자를 삭제하기 위하여 필요한 선택항목을 선택하고 Accept를 찰칵한다.

그룹관리

사용자와 그룹을 관리하는 두번째 양상은 그룹으로 작업하는 경우이다. 그룹관리로 절환하기 위하여 Config표쪽을 선택하고 Linuxconf창문의 왼쪽 면에서 Users Accounts➡Normal➡Group Definitions을 선택한다. 그룹을 관리할 때에는 Group목록과 그림 15-13에 있는것과 같은 기능단추렬로 작업한다. 그룹에 대해서는 제17장 《일반체계관리》에서 상세하게 론의된다.

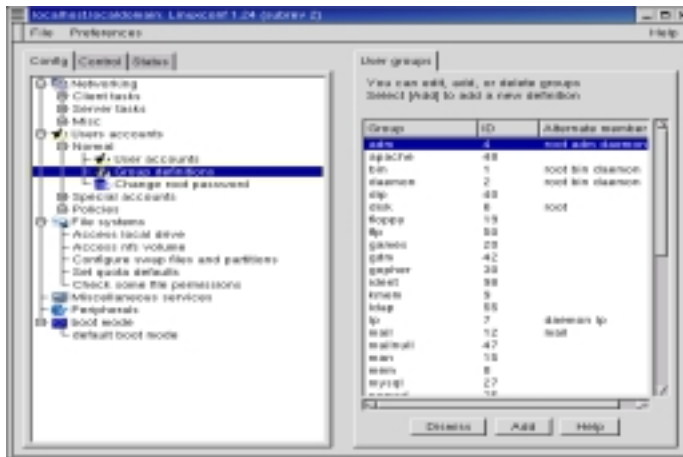



그림 15-13. 그룹관리

그룹추가

group목록에 새 그룹을 추가하기 위하여 간단히 Add를 찰칵한다. 그러면 그림 15-14에 있는것과 같은 화면이 나타난다.

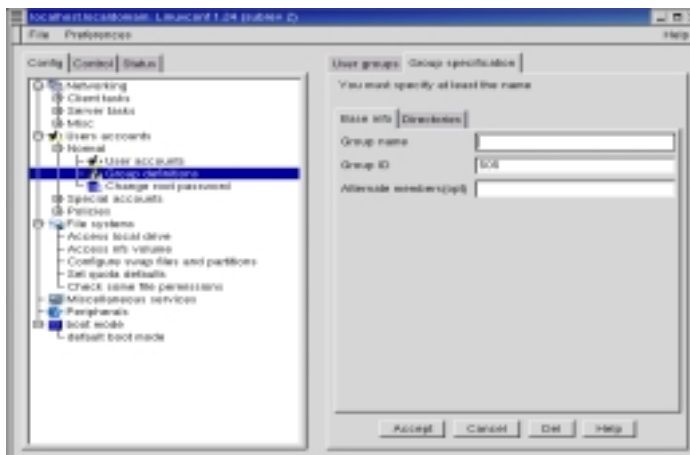



그림 15-14. 그룹의 추가

이것은 체계에 사용자를 추가하는데 리용된 화면의 축소판과 같이 보인다. 여기서는 다음과 같은 기능을 수행할수 있다.

- 그룹이름을 지정한다.
- 체계가 제공한 기정그룹과 같지 않으면 그룹에 대한 GID를 선택한다.
- 매 사용자사이의 공간을 가진 Alternate Members(대리성원)마당에 사용자목록을 입력하여 대리사용자를 그룹의 성원으로 선정한다.

그룹의 편집과 보기

그룹과 관련된 상세한 내용을 편집하거나 보기 위하여 간단히 그림 15-15에 있는것과 같은 창문을 얻을수 있는 기본그룹관리화면의 Group목록에서 그룹을 찰칵한다.

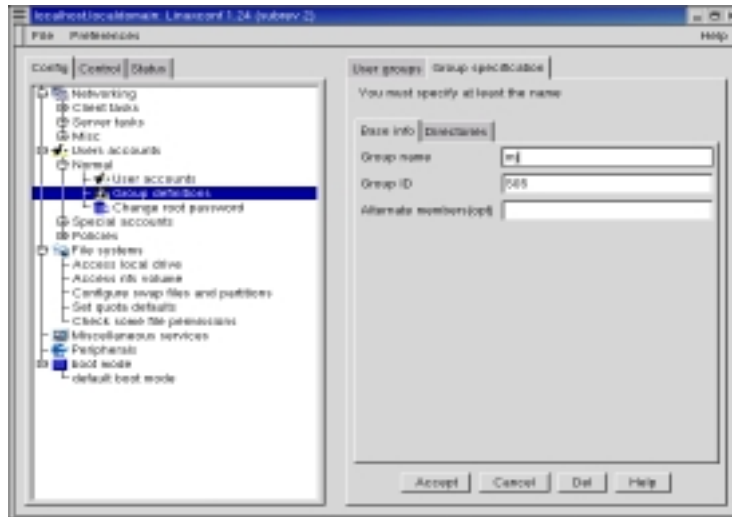



그림 15-15. 그룹의 편집과 보기

화면칸은 여기에 모든 자료가 가득 찬 경우를 제외하고는 그룹를 추가했을 때와 같이 보인다는것을 명심해야 한다. 비록 그룹에 대한 GID나 사용자목록을 바꿀수 있어도 특수하게 행정그룹(GID번호가 100이하)에 대해서 정확히 무엇을 하고 있는지 모르는 한 이것을 할수 없다.

보관 및 끝내기

Linuxconf로 임의의 구성과제를 끝낸 다음에는 변화들을 동작시켜야 한다. 그렇게 하기 위하여 File➡Act/Change를 선택한다. 임의의 추가변화들이 요구되면 그림 15-16에 보여 준것과 비슷한 확인화면이 나타난다. 이 화면은 Linux체계파일에 변화들을 기록하기 위해 어떤것들이 수행되여야 하는가를 보여 준다.

확인화면에서 다음과 같이 선택할수 있다.

- Do It를 찰칵하여 만든 변화들을 적용한다.
- 또는 Do Nothing단추를 찰칵하여 Linuxconf로부터 탈퇴할 결심을 취소한다.

이제는 File➡Quit를 선택해서 Linuxconf를 끝낼수 있다.

모뎀의 구성

역시 Linuxconf를 통해서 모뎀도 구성할수 있다. 이 장에서 이미 본 지령들을 리용하여 Linuxconf를 펼친다. 앞에서 본것처럼 modemconf모듈을 추가했으면 이제는 모뎀을 구성하기 위하여 Linuxconf를 리용할수 있어야 한다. 그림 15-17에서 보여 준것과 같은 Modem Configurator (모뎀구성관리기)를 호출하는 Peripherals➡Modem을 Config표쪽에서 선택한다.

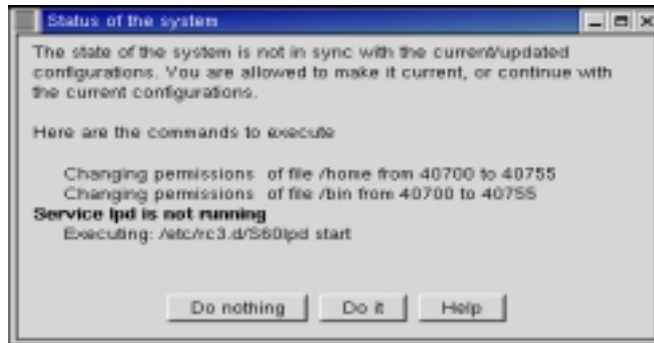



그림 15-16. Linuxconf 변경들의 활성화

이 창문은 선택하는데 극히 적은것을 제공한다. 즉 내부모뎀인 경우에 모뎀의 리용을 위해서 접속되거나 구성된 포구를 선택할수 있다. 이것은 장치 /dev/modem에 있는 모뎀을 제외한 프로그람이 자기의 모뎀을 찾을수 있도록 필요한 련결프로그람을 작성한다.

대신에 모뎀이 접속되는 포구를 검출하는 Linuxconf를 가질수 있다. 모뎀을 전화선에 련결한 다음 Detect를 찰칵한다. 자기의 모뎀이 검출되면 포구를 포함하는 그 효과에 대한 통보문을 보게 되며 반대인 경우에는 No Modem Detected(모뎀이 검출되지 않음) 통보문을 보게 된다.

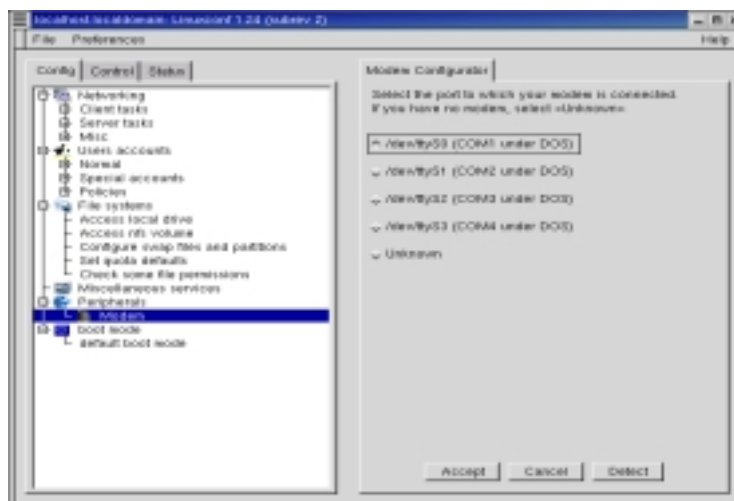



그림 15-17. Modem Configurator

이 공정은 인터네트를 호출하거나 모사전송하는 모뎀을 리용하는 프로그람의 구성에 관계 없다. 이러한 프로그람들은 제22장 《Linux를 인터네트에 접속》과 같은 뒤장들에서 보게 된다.

일러두기 Linuxconf 모뎀구성관리기가 모뎀을 검출할수 있으면 발신음을 내는 전화선에 모뎀이 접속되였는가 확인한다. 대신에 제22장에서 보게 되는 도구를 리용하여 Linux콤퓨터가 모뎀을 검출하는것을 방조할수 있다.

시간과 날자의 설정

역시 Linuxconf를 통해서 자기 콤퓨터에 날자와 시간을 구성할수 있다. 이 장의 앞에서 본 지령들을 리용하여 Linuxconf를 펼친다. Control표쪽에서 그림 15-18에 보여준것처럼 Workstation Date & Time 구성관리기를 호출하는 Date & Time을 선택한다.

시간과 날자는 표쪽의 중간과 아래부분에 현시된다. 시간이나 날자(시, 분, 초, 월, 일, 년)에 대해서 임의의 부분을 선택할수 있고 변경시킬수 있다.

Linux는 그리니치표준시간(GMT:Greenwich Mean Time)을 리용하도록 설정된다. 이것은 서로 다른 시간지대에 위치하고 있는 Linux 봉사기들의 시간을 쉽게 맞추게 한다. 자기의 콤퓨터로 이 작업을 하기 위해서 Linux는 Zone마당옆에 있는 내림차림표를 찰칵하여 선택할수 있는 지정한 시간대를 알아야 한다.

자기의 콤퓨터에 Linux조작체계만이 있는 경우에는 Store Date In CMOS선택항목을 선택한다. 이것은 지금 자기가 설정하는 GMT판의 시간을 콤퓨터하드웨어시계에 기억시킨다. 한편 사용자의 시간대(즉 일광절약시간을 위하여)에서 시간을 변경할 때에는 Linux시계를 변화시켜야 한다.

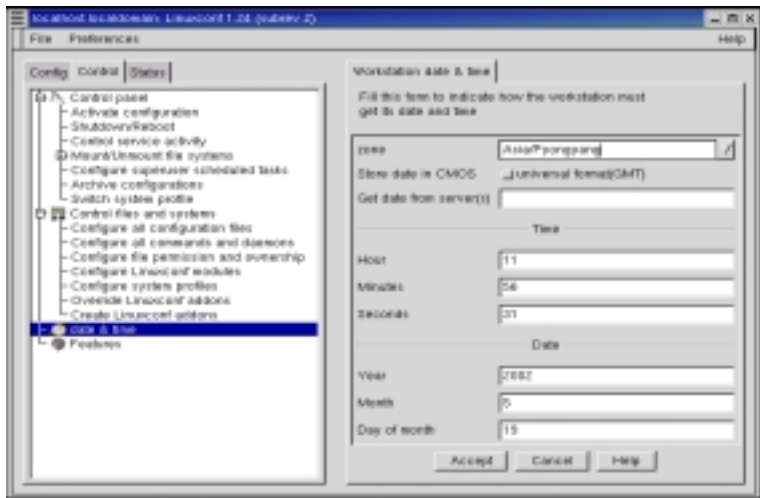



그림 15-18. Workstation Date & Time Configurator

자기 콤퓨터의 시간을 Get Date From Server본문칸에 추가할수 있는 다른 봉사기와 맞추는데 역시 이 도구를 리용할수 있다. 이것은 netdate지령이든지 xntp시간봉사기든지 어느 하나에 기초한것이다. 이 편의프로그람들은 어느것도 Red Hat Linux 7.1에 포함되여 있지 않다. 필요한 변경들이 완성되였으면 Accept를 찰칵하고 이 장에서 본것처

럼 변경들을 동작시킨다.

주의	각이한 콤퓨터에 동시에 필요한 응용프로그람이 있으면 ntpdate 편의프로그람은 매 콤퓨터를 시간봉사기와 동기화할수 있다. 더 구체적인 정보에 대해서는 이 편의프로그람에 대한 Web페지 즉 http://www.eecis.udel.edu/˜ntp/ntp_sp ool/html/ntpdate.htm을 보아야 한다.

▶미리 알아두기

이 장에서는 인쇄기, 모뎀, 사용자, 그룹과 같은 체계의 가장 기초적인 구성요소들 즉 매일 리용되는 구성요소들을 어떻게 구성하는가에 대한 기초지식을 주었다. 제21장《Linux망련결에 대한 리해》에서 취급한 망구성관리기 netcfg와 제31장《보안과 값 눅은 경로기로서 Red Hat Linux 7.1》에서 취급한 방화벽구성관리기인 firewall-config를 비롯하여 기타 련관된 구성도구들은 앞으로 보게 되는 장들에서 고찰한다.

Unix와 Linux파일관리에 대한 지식과 기초적인 리해를 가졌으면 쉘을 깊이 파고들수 있는 준비가 되였다고 볼수 있다. 쉘은 Linux권위자들이 좋아하지만 Linux가 미숙한 사람들이 두려워 하는 지령행을 제공하는 프로그람이다.

제16장《쉘에 대한 리해》에서는 쉘의 기능과 리용할수 있는 많은 쉘(가장 일반적인 Bash쉘과 C쉘들을 포함한다.)들을 비교하고 거의 모든 매개 Linux배포물의 기정쉘인 Bash에 대해 보다 자세히 언급하고 있다.

제16장. 쉘에 대한 리해

지금까지 몇가지 기초적인 Unix지령에 대하여 알게 되였으므로 이제는 그것들을 입력하는데 필요한 지령행환경을 리해하는것이 중요하다. 이 지령행환경은 지령해석기인 쉘로 알려 져 있다. Windows 3.1과 Windows 98의 DOS재촉문과 비슷한 이 쉘은 지령을 입력하고 프로그람을 실행시키며 파일을 조작하게 한다.

DOS재촉문이 제한된 유연성을 가진 고정된 환경인 Windows세계에서와는 달리 Unix쉘은 가입할 때 프로쎄스로 실행되는 작은 응용프로그람이며 각이한 사용자와 응용프로그람에 맞는 다양한 지령행대면부특성과 성능을 제공한다.

Unix세계에는 Linux의 확장자에 따라 선택할수 있는 많은 쉘이 있다. 매 쉘은 각이한 특성과 성능들을 제공하며 대부분은 자기들이 소유한 스크립트언어를 제공하는데 그것들은 정교하고 DOS계렬파일(널리 알려 진 .BAT확장자)과 비슷하지만 그것보다 더 강력한 자체실행되는 프로그람을 작성할수 있게 한다.

이 장에서는 우선 Bourne Shell, C Shell 및 Korn Shell을 비롯하여 주요Unix쉘들의 개념을 주며 다음으로 Bash로 알려 진 Bourne Again Shell을 구체적으로 고찰하게 되는데 Bourne Again Shell은 대부분의 Linux배포물들에 설치된 기정쉘이다.

쉘의 비교

제일 최초의 Unix쉘은 리력목록도 없고 지령행편집기능도 없으며 일감조종기능도 없는 아주 제한된것이였다. 이것은 쉘들이 Unix의 계에서 관심으로 되였던 1970년대의 Bourne Shell과 C Shell의 출현에 의한것이였다. 오늘날 Unix사용자들을 위한 쓸모 있는

과도적인 쉘들이 있는데 이것들은 서로 구별되는 여러가지 고유한 특성을 가지고 있다.

쉘에는 두가지의 기본부류 즉 Bourne Shell로부터 기본문법을 끌어 내여 설계된것과 모형이 C Shell에 기초한것이 있다. 거의 모든 Linux배포물과 함께 제공되는 기정쉘인 Bourne Again Shell을 구체적으로 고찰하기전에 먼저 일부 주요Unix 쉘을 먼저 고찰하자.

일러두기	만일 Unix쉘세계의 가장 최근 발전에 대하여 보려면 comp.unix.shell새소식그룹을 읽어 보아야 한다.

Bourne Shell(sh)

많은 사람들은 Bourne Shell(sh)을 초기Unix 쉘로 보고 있다. 많은 경우에 그것은 오히려 리력목록과 지령행편집기능과 같은 일련의 특성들이 부족하거나 제한된 편이다. 그러나 그것은 초기Unix쉘이므로 그것의 기본지령묶음을 비롯한 Bourne Shell의 많은 특징들은 대부분의 현대쉘에서 볼수 있다. 그러나 Bourne Shell을 자기의 기정쉘로 쓰는 현대체계를 좀처럼 보기 힘들다. 대신에 거의 모든 Linux배포물들에서는 Bourne Again Shell이나 Enhanced C Shell을 공통적으로 볼수 있다.

Bourne Shell은 탈퇴할 때 프로그람의 성공 혹은 실패상태에 대해 검토하는 능력과 같은 많은 주요쉘의 개념들을 도입한것으로 널리 알려 졌는데 이것은 정교한 스크립트화를 고려한다. 이 개념은 지금 모든 Unix Shell에 확립되였다.

C Shell(csh)

C Shell(csh)은 일반적인 C프로그람언어의 문법으로부터 파생된 지령과 스크립트화 환경을 제공하기 위해서 개발되였던 초기의 Unix 쉘이다. 초기의 Bourne Shell과 같이 C Shell은 지령행편집기능과 같은 중요한 몇가지 특성이 부족하였다. 아직도 C Shell은 지령별명 및 지령리력을 비롯한 많은 주요착상을 받아 들이고 있다. 지령행편집기능과 Bash에서 볼수 있는 다른 특징들을 추가한 Enhanced C Sell(tcsh)은 보통 대리쉘로서 거의 모든 Linux배포물에 설치된다.

C Shell은 오늘날 Bourne Shell에 기초한 쉘들에서도 볼수 있는 몇가지 중요한 개념을 도입하였기때문에 일반적으로 환영을 받고 있다. 이것들은 산수계산 및 비교검토와 같은 일정한 기능들을 쉘자체로 수행할수 있다는 사상을 포함한다. Bourne Shell인 경우에 이런 과제들은 외부프로그람들의 호출을 요구한다.

Bourne Again shell(bash)

Bourne Again Shell(bash)은 Linux배포물에 설치된 가장 공통적인 쉘이다. Bash로 알려 진 이 쉘은 Bourne Shell에 기초 (이름이 나타내는것처럼) 하고 있지만 지령행편집, 리력목록 및 파일이름완성기능을 포함하는 폭 넓은 추가특성을 제공한다. Bash가 있으면 Bourne Shell과 같은 문법을 리용한 정교한 쉘스크립트묶음을 리용할수 있다. 가장 공통적인 Linux 쉘로서 Bash는 이 장에서 기본초점으로 되며 이 책의 모든 실레에 리용하고 있는 기정쉘이다.

Korn shell(Ksh)

Korn Shell(ksh)은 초기의 Bourne Shell로부터 파생된 쉘계렬의 또 다른 쉘이다. 몇가지를 고려하면 Korn Shell은 가장 일반적으로 리용하는 Unix 쉘이지만 거의 모든 Linux체계들에서의 기정쉘은 아니다. Korn Shell은 아마도 지령행편집기능을 비롯하여 Bash에서 본 일반적인 많은 특성을 받아 들인데서 첫 자리를 차지할것이다. Korn Shell은 또한 C Shell에 도입된 많은 특성들을 Bourne Shell세계에 가져다 준 첫 쉘들중의 하나였다.

기타 쉘

모두가 자기자체의 고유한 지령형대면부를 제공하는 많은 또 다른 쉘들이 있다. 그러나 이것들은 그리 일반적이지도 못하고 대부분의 Linux배포물에 태울수 있도록 잘 알려 지지도 않았다. 이 쉘들에는 Bourne Shell 부분모임인 Adventure Shell(ash), 프로그람을 완전히 다시 짤수 있는 쉘환경을 제공하는 Extensible Shell(es), 쉘에 보다 경제적이고 명확한 구조를 가진 취급법을 제공하는것을 목적으로 하는 쉘인 ERGO Shell(esh) 그리고 Bash와 같이 여러가지 쉘로부터 특징들을 수집하며 Korn Shell과도 비슷한 Z Shell(zsh)등이 속한다.

일러두기 Linux체계를 위한 특수한 대리쉘에 대한 원천은 SunSite Linux문서고에서 볼수 있다. 그것들의 쉘등록부 ftp://metalab.unc.edu/puh/Linux/system/shells/에서 lsh, pash 그리고 pdksh와 같은 그리 공통적으로 쓰이지 않는 쉘의 묶음뿐아니라 거기에 렬거된 모든 쉘들을 볼수 있다.

각이한 쉘에 대한 시험

이 시점에서는 아마 각이한 쉘들을 기정쉘로 만들지 않고 그것들을 시험해 볼수 있겠는지 알수 없다. 기정쉘안의 프로그람으로 쉘을 간단히 실행하여 이것을 할수 있다. 례를 들어 Enhanced C Shell은 Bash가 모든 사용자들에게 기정쉘로 되도록 설정하여도 초기설치된다. 만일 tcsh를 시험해 보려면 다음 지령을 입력해야 한다.

```
$ tcsh
```

이 지령은 tcsh가 경로우에 있다고 가정한다. 만약 없다면 /bin/tcsh와 같이 그 쉘에 적합한 경로를 포함시켜야 한다. 일단 이것을 집행하면 Enhanced C Shell은 실행되며 그 쉘에로 절환된다. 간단히 exit를 입력하면 자기의 기정쉘로 복귀된다.

대부분의 Linux체계에서 /bin/sh로부터 /bin/tcsh까지의 기호련결프로그람을 찾을수 있으므로 다음의 지령을 주어도 역시 Enhanced C Shell이 기동된다.

```
$ csh
```

만일 각이한 쉘들을 시험해 보려면 자기의 기정쉘을 절환하기전에 그것들을 모두 시험하기 위해서 이 수법을 쓸수 있다. 자기의 체계에 가입할 때 실행하는 기정쉘을 변경하기 위해서는 Unix통과암호파일의 기입사항을 바꾸어야 한다. chsh지령을 리용하여 이 작업을 할수 있다.

기정쉘이 현재 /bin/bash(Bourne Again Shell)인 사용자 someuser를 보기로 하자.

Unix통과암호파일에서 그것들의 현재기입사항은 다음과 같다.

```
someuser::790:103:some User:/home/someuser:/bin/bash
```

이 통과암호기입사항에서 마지막부분은 쉘을 /bin/bash로 지정한다. 다음의 지령을 실행하면 someuser에 대한 쉘을 /bin/tcsh로 변경시킨다.

```
$ chsh -s /bin/tcsh someuser
```

결국 Unix통과암호파일에서 someuser에 대한 기입사항은 다음과 같다.

```
someuser::790:103:some User:/home/someuser:/bin/tcsh
```

그러면 다음번에 그것들이 가입한다. Enhanced C Shell은 Bourne Again Shell대신에 그것들의 쉘로 될수 있다.

Bash shell의 개괄

이제는 Linux의 가장 공통적인 쉘인 Bash Shell를 더 구체적으로 고찰하자. Bash Shell이 Linux세계의 어디에 적합한가를 배우는 첫 단계로 전반적가입과정을 리해하는것이 중요하다. Login에 가입할 때 몇가지 사건이 일어 난다. 그 첫번째는 쉘(이 경우는 Bash)의 기동이고 다음은 자기의 개인적인 Bash환경을 위하여 작성하여야 했던 임의의 구성파일에 대한 실행이다.

Bash에 개성화된 구성을 제공하기 위하여 홈등록부에 .bashrc라고 하는 파일을 작성해야 한다. 이것은 쉘을 기동할 때 즉 일반적으로는 가입할 때 언제든지 Bash에 의해 실행되는 간단한 본문파일이다.

주의	Red Hat Linux 7.1에서 개성화된 구성선택항목은 역시 **.bash_profile**라는 파일에 포함되며 역시 홈등록부에 있다.

.bashrc파일에는 정교한 스크립트화지령은 물론 정상적인 Bash재촉문에 입력해야 할 지령들이나 Bash기능들에 대한 임의의 알맞는 조합이 포함될수 있다.

그것은 Bash의 동작을 구성하고 경로와 같은 환경변수를 설정할수 있는 .bashrc파일에 있다. 그리고 Bash 쉘을 실행할 때마다 필요한 임의의 프로그람을 기동할수 있다. 다음의 지령들은 전용화화된 재촉문을 작성하고 which라고 하는 지령별명을 설정하며 일부 환경변수를 할당하는 .bashrc파일을 만든다.

```
PS1="[\ u@\ h\ w]\ \ $"
alais which="type -path"
export PATH=$PATH:.:~/bin
export EDITOR=emacs
```

걸보기에는 작은 실례의 기능에는 Bash쉘에 대한 여러가지 주요특징들이 있는데 여기에는 다음의것들이 포함된다.

- 환경변수의 설정
- 지령별명
- 문양확장

이 모든것은 Bash쉘에 대한 다음의 주요특징들과 함께 이 장의 나머지부분에서 보게 된다.

- 입출력방향바꾸기
- 파일이름완성
- 완전지령행편집
- 지령리력목록
- 일감조종

Bash의 환경변수설정

매개 쉘은 환경변수를 설정하는 자체의 문법을 가진다. Bash에서 이것은 보통 두 단계 즉 변수값설정과 변수를 환경에 반출하기로 진행된다. 례를 들어 기정본문편집기로 emacs를 설정하려면 다음의 지령으로 EDITOR환경변수를 설정할수 있다.

$ EDITOR=emacs

그리고 다음지령으로 그것을 반출할수 있다.

$ export EDITOR

이 두 단계는 한개 지령 즉 EDITOR 에 값을 배정하고 EDITOR를 반출하는 한개의 단계로 조합할수 있다. 즉

$ export EDITOR=emacs

문양확장을 론의할 때 보다 구체적으로 배울수 있겠지만 Bash지령에서 변수이름의 시작에 $를 추가하여 환경변수값을 호출할수 있다. 이 방법으로 정보를 환경변수의 현재값에 추가할수 있다. 례를 들어 PATH변수에 현재

/bin:/usr/bin:/usr/X11R6/bin

이 있으면 다음의 지령을 리용해서 경로에 /usr/local/bin을 추가할수 있다.

$ export PATH=$PATH:/usr/local/bin

이 지령이 PATH변수의 현재 값을 PATH변수의 새 값에 포함시키기 위해 $PATH를 어떻게 리용하는가에 주목을 돌려야 한다. 변수값은 결국 다음과 같이 된다.

/bin:/usr/bin:/usr/X11R6/bin:/usr/local/bin

입출력방향바꾸기

Unix의 세계에서 가장 쓸모 있는 개념들중의 하나는 표준입력과 표준출력에 대한 개념이다. 흔히 호환되지 않는 프로그람들은 자기들의 입력자료를 표준입력 즉 보통 건반을 통해서 입수한다. 마찬가지로 그것들은 결과를 표준출력 즉 보통 화면에 현시한다.

그러나 거의 모든 쉘들은 표준입력과 표준출력의 방향을 바꾸는 능력을 제공하는데 이것은 다중지령들과 자료파일들의 복합지령조합을 만들수 있게 한다. 제 15장에서 본것처럼 막대기(|)를 쓰면 표준출력을 표준입력으로 방향바꾸기할수 있다. 례를 들어 지령

$ ls -1 | more

으로는 ls -1지령의 출력이 표준출력에 현시되지 않지만 대신에 그것을 more지령의 표준입력으로 바꿀수 있다. more지령은 바꾸어 진 이 입력을 처리하며 결과를 표준출력에 현시한다.

Bash에는 표준입력과 출력을 바꾸는 또 다른 방법 즉 〈and〉방향바꾸기기호들을 리

용하는 방법이 있다. 례를 들어 등록부를 후에 처리하는 파일로 목록화하여 보관하려고 하는 경우를 보자. 표준출력을 바꾸기 위해서 >기호를 쓸수 있다. 즉

```
$ ls -1 > filelist
```

이 지령은 filelist라고 하는 파일에 ls -l지령의 출력을 보관한다. 마찬가지로 만일 지령에 대한 표준입력으로 파일의 내용을 리용하려면 <기호를 리용해야 한다. 즉

```
$ mail user@juxta.com < filelist
```

이 지령을 리용하면 filelist파일의 내용을 mail지령을 위한 입력자료로 리용하고 효률적으로 파일의 내용을 전자우편통보문의 본체로 리용할수 있다.

파일이름완성

때때로 단어완성이라고 하는 파일이름완성은 단순하지만 대단히 쓸모 있는 개념이다. 이것은 다음과 같이 작업한다. 즉 파일, 지령 또는 등록부이름을 유일하게 정의하기 위해서문자들을 충분히 입력하면 Bash는 이름의 나머지부분을 완성할수 있다. 다음과 같이 입력하였다고 하자.

```
$ /usr/lo
```

거의 모든 체계에서 /usr/lo는 등록부 /usr/local을 유일하게 정의한다. 만일 간단히 Tab건을 누르면 Bash는 이름을 완성하도록 한다. 한개의 파일, 지령 또는 등록부가 /usr/lo로 시작하면 Bash는 사용자에 대한 이름을 완성하며 이런 경우에는 /usr/local을 읽도록 본문을 적어 넣는다.

물론 이름을 완성하는 방법을 정통할수 있는 Bash에 대한 충분한 정보가 제공되지 않으면 시간이 걸린다. 이런 상황에서 쉘은 완성하려는 이름을 일치시키기 위해서 모든 가능한 대리안들을 제공할수 있다. 례를 들어

```
$ /usr/1
```

을 입력하면 /usr/local과 /usr/lib/가 보통 일치한다. 만일 Tab를 누르면 Bash는 혼돈된다. 즉 어느 이름을 완성하겠는지 모르므로 그것은 이름을 완성할수 없다. 만일 Tab를 두번 누르면 Bash는 모든 가능한 대리안들을 현시한다. 즉

```
lib    local
```

필요한 이름을 유일하게 정의하기 위해서 충분한 문자들을 자유로 입력한 다음 Tab를 다시 누른다. 우에서 본 실례에서 추가o를 입력하고 Tab를 누르면 Bash는 이름을 /usr/local/로 완성한다.

지금까지는 대체로《만일 파일에 완전한 경로를 제공하지 않고 국부등록부나 경로의 임의의 곳에 파일이름을 입력하려고 하면 어떤 현상이 일어 나겠는가? 그러면 Bash는 파일이름을 완성할수 있는가?》라고 질문하면 《예》라고 대답하였다. 만일 파일이나 등록부의 이름을 완전한 경로를 제공하지 않고 완성하려는 경우

```
$ gr
```

에 Bash는 입력된 문자들과 일치하는 이름들의 경로를 탐색한다. 대부분의 체계들에서 이미 본 실례는 grep와 groff를 비롯하여 몇개의 지령들만을 정합시키며 Bash는 가능한 대리안들의 목록을 제시하게 된다.

```
grep groff grotty
```

만일 지령 grep를 리용하려면 e를 입력하고 Tab를 다시 눌러야 한다. 그러면 Bash는 지령이름을 완성한다.

지령별명

지령별명짓기는 사용자자체의 전용지령을 정의할수 있게 하는 Bash 쉘의 강력한 특성이다. 례를 들어 체계에서 실행되는 모든 공정들을 정규적으로 검사하려면 다음과 같은 지령을 리용해야 한다.

```
$ ps aux | more
```

체계에서 실행하는 모든 공정을 검사하려고 할 때마다 **ps aux | more**를 입력하지 않고도 맞추어 넣은 Bash의 alias지령을 리용하여 별명을 정의함으로써 불필요한 건조작들을 피할수 있다.

이 실례에서 실행하는 모든 공정들을 검사하려고 할 때마다 리용하는 psa라고 하는 별명을 작성하려고 한다고 하자. 다음의 지령을 리용해서 이 별명을 정의할수 있다.

```
$ alias psa="ps aux | more"
```

일단 이것을 수행하면 Bash에서 **psa**를 입력하여 지령 ps aux |more를 실행한다. 이 별명은 현재 Bash대화조종의 마지막까지 지정한 자리에 남아 있다. 가입에서 탈퇴하고 Bash가 탈퇴하면 지정되였던 별명은 없어 진다.

만일 영구적인 지령별명을 작성하려면 기정본문편집기 emacs를 리용할수 있는데 이미 쓰던 방법대로 별명을 자기의 .bashrc 파일에 설정해야 한다.

일반적으로 리용된 지령에 대한 간략이름을 작성하는 방법을 제공하는것외에도 지령별명짓기는 큰 실수를 방지하는데도 리용될수 있다. 례를 들어 뿌리사용자로 가입하면 cp, mv 및 rm지령을 정상적으로 리용하는것은 아주 위험할수 있다.

어떤 실수는 그렇게 하지 않았음에도 불구하고 파일, 등록부 또는 전체 파일체계를 덧쓰거나 지워 버릴수 있다. 이러한 원인으로 해서 뿌리회계의 Bash쉘은 다음과 같은 기정별명묶음을 가지도록 구성된다.

```
alias   cp='cp -i'
alias   mv='mv -i'
alias   rm='rm -i'
```

이런 모든 경우에 –i기발은 지령으로 하여금 파일을 덧쓰거나 지우게 하는 모든 동작에 대하여 사용자에게 서로 응답하도록 재촉한다. 매 동작에 대한 입력재촉이 없이 지령이 실행되게 하자면 사용자는 이에 알맞는 기발들(례를 들어 rm, -f)을 설정하도록 정확히 요구해야 한다.

지령행편집

Bourne Shell 및 C Shell과 같은 일부 초기의 쉘에서 빠졌던 Bash의 한가지 특성은 지령행을 편집하는 능력이다. 이 기능이 없이 일단 지령행에 문자렬들을 입력하면 그 문자렬로 자유롭게 돌아 갈수 없다. 즉 오히려 편집하려는 렬의 뒤에 놓이는 모든 문자들을 지우고 문자를 결정한 다음 행의 나머지부분을 다시 입력하여야 한다. 례를 들어 지령

```
$ /usr/kocal/bin/mycommand
```

을 입력하고 다음에 kocal이 local이 되도록 하려면 k의 뒤에 있는 문자들을 삭제하고 k를 l로 바꾼 다음 행의 나머지부분을 다시 입력해야 한다. 두말할것없이 이것은 작업에서는 능률이 높지 못한 방법이다.

이런 리유로 해서 Bash를 비롯한 거의 모든 현대적인 쉘들은 완전한 지령행편집기능을 제공한다. 이 기능이 있으면 현재 행을 훑으면서 움직이는 방향건을 리용하고 필요에 따라 문자를 삭제 및 삽입하며 행끝에 유표를 이동하지 않고 지령을 실행하는 Enter건을 누를수 있다. 기정으로 Bash는 복귀되는 삽입기능을 가지므로 새 문자를 입력하면 그것들은 현재 문자를 덧쓰지 않고 유표위치에 삽입된다.

Bash는 특별히 긴 지령행에서 편집속도를 높이기 위해 몇가지 쓸모 있는 지름건들을 제공한다. 표 16-1에는 이 지름건들을 주었다.

표 16-1. 지령행편집에 쓸모 있는 지름건

건	동 작
Ctrl+A	행시작에로의 이행
Ctrl+E	행끝에로의 이행
Ctrl+T	유표의 왼쪽에 있는 문자를 현재유표위치에 있는 문자로 바꾸기
Esc,T	유표의 왼쪽에 있는 단어를 현재유표위치에 있는 단어로 바꾸기
Esc,U	현재단어를 대문자로 고치기
Esc,L	현재단어를 소문자로 고치기
Esc,C	현재단어를 대문자로 표시하기
Ctrl+K	현재유표위치로부터 행끝까지 삭제하기

만일 emacs편집프로그람을 써보았다면 이 표의 많은 건들이 emacs에서 리용한것들과 같다는것을 알수 있다. 그것은 Bash의 기정편집방식이emacs 와 같은 지령을 쓰도록 설계되였기때문이다. 대리안은 vi와 같은 방식을 포함하지만 vi와 익숙되고 잘 알수 없는 양식의 편집기가 보통 프로그람작성자와 현대 체계관리기에게는 제한되므로 이 책에서는 vi편집방식을 리용하는 방법을 설명하지 않았다.

표 16-1의 지령목록은 Bash에서 쓸수 있는 모든 편집지령의 극히 일부만을 제시한것이다. 이것은 Bash에 대한 유능한 사용자가 되려는 거의 모든 사용자들에게 필수적인 편집기능의 기본묶음을 준다. 만일 Bash의 편집기능에 대해서 더 깊이 파고 들려면 쉘재촉문에 **man bash**를 입력하여 볼수 있는 bash사용설명서페지의 READLINE부분을 읽어야 한다.

지령리력목록

지령리력목록에 대한 지식은 실제로 아주 단순하다. 실행하는 매개 지령(Ctrl+C대신 Enter를 눌러서)은 가장 최근에 수행된 지령부터 시작하여 가장 낡은것으로 끝나는 반대순서로 호출할수 있는 리력목록완충기억기에 추가된다.

리력목록을 호출하기 위한 가장 간단한 방법은 조사하고 있는 지령을 찾을 때까지 리력목록을 훑어 보면서 웃방향건을 반복하여 누르는것이다. 지령목록은 대체로 두가지 목적 즉 잘못 입력된 지령을 결정하고 대화과정에 여러번 지령을 반복실행하는데 리용된다.

리력목록의 첫번째 응용프로그람을 주소화 해보자. 정확히 말해서 다음지령을 입력

하여 testfile파일의 내용을 보려고 하였다고 하자.

$ moer testfile

Enter건을 누르면 다음행들에 오유통보문이 발생한다. 즉

bash: moer: command not found

지령을 빨리 수정하고 실행하기 위하여 마지막지령을 다시 호출하는 웃방향건을 주사할수 있으며 우에서 본 지령행편집기능을 리용해서 moer를 more로 변경하고 파일을 보기 위해 Enter를 누른다.

마찬가지로 설치하고 있는 프로그람의 구성에 대하여 시험하는 경우를 보기로 하자. 구성파일을 편집하고 프로그람은 여러번 실행하는 공정을 반복하려고 할수 있다. 이 두 단계의 공정에서 매 단계를 반복하여 수행하는데 리력목록을 리용하여 이 과정을 진행할 수 있다.

리력목록의 지령을 호출하는 빠른 방법은 !기호를 리용하는것이다. 리력목록의 지령에서 첫 몇개의 문자가 뒤에 놓이는 !를 입력하여 지정된 문자로 시작하는 가장 최근의 지령을 실행할수 있다.

례를 들어 지령

$ ps aux | grep httpd | more

을 이미 리용했고 같은 지령을 다시 실행하려면 지령

$!ps

을 입력하여 ps aux | grep httpd | more가 리용된 후에 다른 ps지령이 리용되지 않는한 이미 있는 지령을 실행해야 한다. !ps를 리용하면 리력목록에서 ps로 시작되는 가장 최근의 지령을 실행한다.

지령행 편집에서와 같이 지령리력목록으로 더 개선된 기능들을 실행하기 위한 몇가지 개선된 건조작방법이 있다. 표 16-2는 이런 건들을 보여 준다.

표 16-2. 지령리력건조작방법

건	동 작
Ctrl+P	리력목록에서 앞의 지령에로 이동
Ctrl+N	리력목록에서 다음 지령에로 이동
Esc,<	리력목록의 웃지령에로 이행(최초의 지령)
Ctrl+R	리력목록을 따라 반대로 탐색
Esc,>	리력목록의 끝에로 이행(가장 최근의 지령)
Ctrl+O	편집 및 실행하는 리력목록에서 현재 행을 실행하고 다음행을 꺼내기

여기서 가장 흥미 있는것은 Ctrl+R이다. 이것은 리력목록에 대한 거꿀탐색을 동적방식 즉 대화작용방식으로 진행하게 한다. 지령을 입력하기 시작하면 Bash는 지금까지 입력된것과 일치하는 가장 최근의 지령을 현시한다. 더 입력하면 할수록 조사하고 있는 지령들에 대한 탐색범위는 더 좁아 진다.

례를 들어 이미 한번 실행했던 정교한 find지령을 실행하려고 한다고 하자. 거꿀탐색을 시작하기 위해 Ctrl+R를 리용할수 있다. 이것을 진행하면 Bash는 대화방식탐색의 재촉기호를 제시한다. 즉

(reverse-i-search)`':

find지령에 문자를 입력하면 Bash는 현재 입력된 문자렬과 일치하는 가장 최근의 지령을 찾는다. 례를 들어 정확히 **f**를 입력하면 다음과 같은것이 나타나야 한다.

(reverse-i-search)`f': pico info/.signature

계속하여 **fi**를 입력하면 다음과 같은것이 작성되여야 한다.

(reverse-i-search)`f': rm -rf StarOffice-3.1

마지막으로 fin을 입력하면 비로소 조사하고 있는 지령

(reverse-i-search)`f': find / -name 'foo' -print

이 현시된다. 이제 그 지령을 실행하기 위하여 Enter를 누를수 있다.

일감조종

일감조종에 대한 개념은 조작체계의 다중과제처리능력을 주는Linux에서 특별히 쓸모 있다. 일감조종기능을 리용하면 동시에 실행하는 다중프로그람을 실행하고 조종하는데 한개의 쉘을 리용할수 있다.

보통 지령을 실행할 때 그것은 전경에서 실행한다. 즉 쉘은 지령을 실행하고 재촉문은 지령이 완료될 때까지 복귀되지 않는다. emacs와 같은 대화방식프로그람인 경우 이것은 프로그람이 쉘이 실행하고 있는 화면이나 창문을 넘겨 받고 퇴장할 때만 프로그람이 지령재촉기호를 다시 쓸수 있게 된다는 뜻을 포함한다. find와 같은 비대화방식프로그람인 경우 프로그람은 실행되며 그것이 화면에 아무것도 현시하지 않아도 지령재촉기호는 프로그람이 완료될 때까지 복귀되지 않는다.

다음 지령을 고찰하자.

$ find / -name. '*.tmp' -print > templist

이 지령은 .tmp확장자를 가진 파일에 대한 Linux체계의 전체 등록부구조를 탐색하고 표준출력에 그 파일이름들을 모두 표시한다. 앞의 지령에서 출력은 후에 처리하는데 리용할수 있는 templist라고 하는 파일로 바꾸어 진다. 이것은 현시장치에 아무것도 인쇄되지 않는다는 뜻을 포함한다. 즉 find를 실행하고 있으면 다른 지령을 실행할수 없게 한다.

이와 같은 제한성은 다중과제처리에 대한 순수한 개념에 불리하다. find지령을 시작하고 find가 실행을 완료하면 다른 동작을 계속하는 수단이 있어야 한다.

이것은 배경에서 프로그람을 실행하는 착상이 작용하기 시작한 곳이다. 그것을 전경에서 실행하기보다 다른 지령의 동시실행을 금지하기와 일감을 배경에 배치하기는 사용자별지령재촉기호에 새 지령을 넣으면 지령을 실행하게 한다.

일감을 배경에 배치하는 가장 쉬운 방법은 지령을 실행할 때 지령의 끝에 특수문자 &를 추가하는것이다. 이 문자는 Bash가 배경에서 지령을 실행하도록 지시하며 직접 새로운 지령재촉기호를 준다. find지령을 배경에 놓기 위하여 다음지령을 준다.

$ find / -name '*.tmp' -print > templist &

일단 이 지령을 실행하기 위하여 Enter를 누르면 새로운 지령재촉기호에 직접 나타나며 동시에 find가 실행한다.

Bash의 jobs지령을 리용하면 배경에서 실행하는 일감을 추적할수 있다. jobs를 리용하여 다음과 같은 결과들을 만들수 있어야 한다.

$ jobs
[1]+ Running   find / -name '*.tmp' -print > templist &

여기서 jobs지령은 한개의 배경(&에 주목을 돌리시오.)일감실행이 있다는것과 첫 일감실행이 find / -name '*.tmp' -print > templist &라는것을 나타내는 결과를 알려 준다.

배경에 몇개의 일감실행이 있으면 jobs지령으로부터 다음과 같은 결과를 얻게 된다.

[1]- Running   find / -name '*.tmp' -print > templist &
[2]+ Running   ls –lR / >dirlist &

이미 전경에서 프로그람을 실행하기 시작하였고 그것을 배경에 놓으려고 한다면 그렇게 할수 있다. 건조작 Ctrl+Z는 Bash에 의해서 현재 처리공정을 림시 중지하도록 요구하는것으로 해석한다. 례를 들어 앞의 실례에서 find지령이 전경에서 실행되고 있고 Ctrl+Z를 입력하면 처리공정은 림시 중지된다. 이때 jobs지령을 입력하면 다음과 같은 결과가 얻어 진다.

[1]+ Stopped   find / -name '*.tmp' -print > templist

일감의 상태가 Running이 아니라 어떻게 Stopped되는가에 주목을 돌려야 한다. 그러면 bg지령으로 배경에 있는 일감을 배치할수 있다. 즉

$ bg 1

여기서 1은 일감의 번호를 지적한다. 한개의 일감만이 중지되였다면 일감번호는 지적하지 않아도 된다. 즉

$ bg

는 배경에 한개의 중지된 일감을 배치할수 있다. 일단 중지된 일감이 배경에 배치되였으면 **jobs**를 입력하여 Running과 같은 일감을 보여 주는 결과를 얻을수 있다.

때때로 배경에 일감을 배치하지 않고 그것을 림시 중지하는것이 쓸모 있다. 이것은 emacs와 다른 본문편집기와 같은 대화방식응용프로그람을 리용하고 있고 하나이상의 지령을 실행하려는 때 특히 의의 있다. 편집기에서 완전히 탈퇴하는것보다 편집기일감을 중지하는 Ctrl+Z를 리용하고 요구하는 지령을 실행하고 중지된 일감을 전경에 복귀하는것이 더 간편하다. 그러나 그것의 성질로부터 대화방식응용프로그람은 배경에서 실행을 시작하는데 잘 맞지 않으며 그것들이 리용되지 않으면 중지된 그대로 두는것이 더 좋다.

중지된 일감(또는 배경일감)을 전경에 보내기 위하여 fg지령을 리용할수 있다. 례를 들어 emacs를 실행하고 있었지만 Ctrl+Z로 그것을 중지시키고 어떤 지령을 실행하고 다시 emacs로 복귀하려면 다음지령을 리용할수 있다.

$ fg

하나이상의 중지시킨 일감이나 배경일감이 있으면 fg지령을 입력할 때 일감번호를 지정해 주는것이 아주 좋다. emacs지령이 일감2였다면 다음지령은 그것을 전경에 복귀하며 편집을 계속 할수 있게 한다.

$ fg 2

마지막으로 중지된 일감이나 배경일감을 그만두거나 끌내려고 할 때가 있을수 있다. kill지령은 처리공정ID(PID)의 기초우에 있는 지령을 끌내는데 리용할수 있다. 프로쎄스

ID를 찾거나(상세한것에 대해서는 ps사용설명서페지를 보시오.) Bash의 중지된 일감과 배경일감을 직접 끝내기 위해 ps지령을 리용한다.

례를 들어 두개의 배경일감을 가지고 있고 jobs지령을

```
[1]- Running      find / -name '*.tmp' -print > templist &
[2]- Running      ls -lR / >dirlist &
[3]+ Stopped      emacs simefile
```

으로 만들고 kill지령을 리용하여 ls지령을 끝낼수 있으면 다음과 같이 된다.

$ kill %2

PID의 끝내기와는 달리 일감의 끝내기는 퍼센트기호(%)가 일감번호앞에 표시할것을 요구한다는것을 명심해야 한다. 퍼센트기호가 없으면 kill은 현재쉘에 있는 일감 2가 아니라 오히려 2번 PID로 프로쎄스를 끝내려고 한다. 이것은 일감번호와 일감의 PID는 같아 보여도 엄격히 다르기때문에 중요한 차이가 있다.

다음과 같은 실례를 보기로 하자. 즉 이미 쉘에서 실행하고 있는 배경처리공정이 있고 일감2로 새로운 일감을 시작한다고 하자. 이 새로운 일감에 착수할 때 체계는 자동적으로 PID들의 목록에 있는 한 PID에 일감을 배정한다. 이 번호는 100,999 또는 25,678이 되여야 한다. 보통 작은 PID들(특히 100이하의것들)은 Linux가 기동할 때 시작하는 모든 형태의 체계프로쎄스가 다 써버리고 콤퓨터의 전원을 끌 때까지 실행한다. 그러므로 일감의 PID가 일감번호와 일치한다는것은 생각도 할수 없다.

패턴확장자

Bash를 포함한 많은 현대적 쉘의 우점들중의 하나는 하나이상의 지령이나 파일을 지정하는데 쓸모 있는 패턴을 리용할수 있는 능력이다. 개념을 설명하는 간단한 실례를 보기로 하자. 보통 지령

$ ls -1

을 리용하면 현재 등록부의 모든 파일들에 대한 완전한 목록을 얻게 된다. 지정한 파일과 알고 있는 모든것이 파일이름이 문자 z로 시작하는것을 찾으려고 한다고 하자. 그러면 다음지령을 리용할수 있다.

$ ls -1 z*

여기서 z*는 인수로 인정된다. 패턴 z*는 파일이름이 z로 시작되고 그뒤에 령이상의 문자가 놓이는 임의의 파일을 렬거하는것을 나타낸다. 이것은 임의의 DOS나 Windows 사용자가 잘 알고 있는 *와 기능이 같다. 실제로 여기서 진행되는것은 Bash가 z로 시작되는 모든 파일이름의 목록을 작성하고 z*를 이 목록과 바꾸며 특히 모든 파일이름을 ls-1지령에 대한 인수로 간주한다. 그러나 DOS와 Windows의 DOS재촉기호와는 달리 Bash는 간단한 *기호에 리용가능한 더 많은 패턴을 제공한다.

경로이름확장자

문양확장의 한가지 형태는 경로이름확장이다. *를 제외하고 표 16-3에 보여 준것처럼 두가지이상의 다른 주요특수기호를 경로이름으로 리용할수 있다.

표 16-3. Bash의 경로이름확장자

기 호	설 명
?	임의의 단일문자와 정합된다.
[...]	중괄호안에 있는 문자들중의 하나와 정합된다.
[A-F]	첫번째와 두번째 문자사이에 있는 단일문자와 정합된다.
[^...] 또는 [!...]	동일한 괄호안에 지정된것과 서로 다른 어떤 단일문자와 정합된다.
[^A-F] 또는 [!A-F]	첫번째와 두번째문자사이에 있는것과 서로 다른 어떤 단일문자와 정합된다.

몇가지 실례를 보기로 하자. a로 시작되고 z로 끝나면서 이름이 세문자길이가 되도록 모든 파일들을 목록으로 만들려고 한다고 하자. 이때 다음과 같은 지령을 리용할수 있다.

```
$ ls -1 a?z
```

마찬가지로 이름이 간단히 a로 시작되고 z로 끝나는 어떤 파일과 일치되게 하자면 다음과 같은 지령을 리용할수 있다.

```
$ ls -1 a*z
```

차이는 첫번째 실례에서는 파일이름이 세문자길이가 되도록 ?가 정확히 한 문자와 일치된다는것이다. 두번째 실례에서 * 가 령이상의 문자와 정합되는데 이것은 파일이름이 길이가 둘이상인 문자로 된다는것을 의미한다.

다음은 문자 a, b, c, d로 시작하는 모든 파일들의 목록을 보려고 하는 경우를 보자. 이때 다음의 지령을 리용할수 있다.

```
$ ls -1 a* b* c* d*
```

그러나 이 지령은 더 많은 문자들로 시작되는 파일을 보려고 하는 경우에는 오히려 비능률적이고 시끄러운것으로 된다. 이 공정을 보다 쉽게 하자면 다음과 같은 지령을 리용할수 있다.

```
$ ls -1 [abcd]*
```

이 지령은 a, b, c 또는 d가운데서 어느 하나로 시작되고 뒤에 령이상의 문자들이 따르는 임의의 파일들을 목록화한다는것을 의미한다. 그러나 a, b, c 및 d는 문자의 련속적인 렬이기때문에 다음과 같은 지령을 쓰면 보다 간결한 문양으로 만들수 있다.

```
$ ls -1 [a-d]*
```

마지막으로 특수한 문자나 문양을 포함하는 경우와 다른 모든것을 포함하는 경우를 고찰해야 한다. 보관을 목적으로 체계의 모든 홈등록부의 압축된 문서고를 만드는 경우를 보자. 례를 들어 어떤 원인으로 하여 문자 m로 시작되는것들을 제외한 모든 홈등록부의 문서고를 만들려고 한다고 하자. 이때 다음지령을 리용할수 있다.

```
$ tar czvf home.tar.gz /home/[a-1]* /home/[n-z]*
```

그러나 다음지령으로 더 쉽게 만들수도 있다.

```
$ tar czvf home.tar.gz /home/[!m]*
```

이것은 m를 제외한 임의의 문자로 시작하는 모든 홈등록부가 이 지령으로 처리된다는것을 의미한다.

한걸음 더 나아가서 다음지령들을 리용하여 m, n 및 o로 시작하는 모든 등록부를 포함할수 있다.

$ tar czvf home.tar.gz /home/[^mno]*

또는

$ tar czvf home.tar.gz /home/[!m-o]*

대괄호확장

경로이름확장과 밀접히 련관된것이 대괄호확장이다. 대괄호확장은 발생되고 있는 이름들이 실제로 파일이나 등록부로 존재하는가에 전혀 관계없이 표현식을 확장할수 있는 방법을 제공한다. 대괄호표현식의 실례에는 다음과 같은 지령이 있다.

$ mkdir testdir{1,2,3,4}

이것은 등록부 testdir1, testdir2, testdir3및 testdir4를 작성할수 있게 한다.

여기서 대괄호안의 매개 요소는 반점으로 분리되며 이러한 요소들은 하나씩 결과이름을 만드는데 리용된다. 이미 본바 있는 [...]표현식과 달리 반점의 리용은 대괄호사이 요소들이 길이가 한 문자이상 될수 있다는것을 의미한다. 즉

$ mkdir testdir{01,02,03,04}

대괄호확장은 적어도 하나의 반점을 포함해야 한다는것을 명심하는것이 중요하다.

경로이름확화장과 대괄호확장에서 흥미 있는 특징은 그것들이 다른 대괄호표현식에 리용될수 있다는것이다. 이것은 첫번째로 일어 나는 확장이 대괄호확장이기때문에 가능하다. 다음실례를 고찰하자.

$ mkdir newdir/{firstdir,firstdir/dir{01,02}}

이 지령은 다음의 등록부들을 작성한다.

firstdir
firstdir/dir01
firstdir/dir02

여기서는 바깥대괄호표현식안에 대괄호표현식 {01,02}를 리용했다.

마찬가지로 대괄호안에 경로이름표현식기호도 쓸수 있다. 즉 지령

$ chmod 644 testfile.{tx?,bak,0[0-9]}

는 표현식 testfile.tx?, testfile.bak, testfile0 ~testfile9에 일치하는 파일을 비롯해서 련속적인 파일들에 대한 허가를 변경시킨다.

지령교체

마지막으로 문양확장의 다른 형태인 지령교체에 대하여 보자. 일부 방법에서 이것은 모든 관련결지령과 비슷하지만 련결에서 지령의 표준출력이 다른 지령의 표준입력으로 간주된다는 점에서는 차이가 있다. 그러나 지령교체에서 한 지령의 표준출력은 다른 지령에 대한 인수나 파라메터로 된다.

례를 들어 .bak확장자를 가진 모든 파일을 압축하려고 하는 그런 실례를 보자. 다음의 지령을 리용하여 이런 모든 파일의 목록을 얻을수 있다.

```
$ find / -name '*.bak' -print
```

find지령의 exec기발을 리용하여 해당한 모든 파일을 압축할수 있다. 즉

```
$ find / -name '*.bak' -exec gzip { } \ ;
```

기타선택항목은 다음지령에서와 같이 지령교체를 리용한다.

```
$ gzip `find / -name '*.bak' -print`
```

여기서 find지령은 거꿀단일인용부호에 들어 있다. 이 인용부호는 지령의 결과에 따르는 지령행의 부분으로 리용되여야 한다는것을 나타내며 이 경우에는 gzip지령에 대한 인수로 된다.

거꿀단일인용부호를 대신하는것들에는 같은 의미를 가지고 같은 결과를 얻게 하는 다음의 형태들이 있다.

```
$ gzip $(find / -name \*.bak -print)
```

이 두개 형태의 기본차이는 특수문자 \ 에 있다. 다음의 실례에는 특수한 의미를 가지는 문자가 없다. 한편 앞에서 본 실례에는 $,` 또는 \ 가 올 때를 제외하고는 거꿀빗선이 특수한 의미를 가진다.

주의	여기서 론의된 확장의 형태를 제외하고 Bash에서 리용할수 있는 다른 형태의 확장이 있다. 그것들에 대해서는 **bash**표준설명서페지에서 상세히 제시되여 있다. 그러나 이 장의 패턴확장형태들은 적은 능력을 제공한다. 보다 강력한 패턴 및 확장기능을 보려면 지령 **man bash**를 리용하여 사용설명서페지를 검색해야 한다.

▶미리 알아두기

이제는 쉘의 강력한 사용자로 되였고 Bash세계에서 많은 멋들어 진 기능을 수행할수 있게 되였다. 즉 Linux에서 능률적이고 실질적인 응용프로그람으로 작업하는데로 넘어 갈수 있게 되였다.

제17장《일반체계관리》에서는 체계관리의 기본과제들에 대하여 배운다. 여기서는 자동적인 실행에 대한 과제들의 일감작성과 Linux체계를 기동하는 방법을 조종하는것은 물론 사용자와 사용자그룹을 관리하는 문제들을 고찰한다.

제17장. 일반체계관리

이 장에서는 동작하는 Linux체계를 유지하는데 필요한 일련의 기초적인 관리과제들에 대하여 고찰한다.

망봉사기로부터 매 사용자들이 공유한 가정용콤퓨터에 이르기까지 Linux체계에 대한 기초지식은 사용자관리이다. 이것은 새로운 사용자회계의 작성으로부터 사용자암호의 바꾸기, 사용자회계가 만들어 졌을 때 체계관리기로서 사용자홈등록부의 보호에 이르기까지의 모든 기능을 수행한다.

지어 세련된 일반체계에서의 체계자원에 대한 사용자접근은 초기사용자기지들과 초

기그룹기지들에 따라 좌우된다. 여기서 그룹은 단일이름을 가진 공동조직체에 소속된 수많은 사용자들로 이루어 진다. Linux는 사용자들을 그룹에 소속시키고 체계자원에 대한 접근을 관리하는 그룹들을 리용하기 위한 기능들을 제공한다.

Linux의 우점을 충분히 발휘하도록 하는 또다른 한가지 관리는 기동과 일람표기지들에서의 과제들에 대한 자동화이다. 실행준위들에 대한 Linux의 개념은 기동과정(및 완료과정)에 제기되는 문제들을 정의하도록 위력한 기구를 제공한다. 그리고 데몬들을 일람표화하는 Cron은 매일, 매주, 매달 혹은 매해 반복되는 일감들과 조작들을 한번에 일람표화하는 도구를 제공한다.

마지막으로 Linux는 체계에서 발생하는 사건들을 정확히 알수 있게 기록하는 일지작성(logging)능력들을 제공한다. 물론 사용과정에 만들어 지는 기록파일에 대해서 그것들을 규칙적으로 관리할 필요가 있다. 결국 기록파일에 한해의 사용정보가 포함되면 아주 커서 과제 혹은 강력한 과제를 빨리 분석하거나 추적하는데 매우 유용하다.

사용자관리

제15장 《체계구성을 위한 Linuxconf와 기타도구들의 리용》에서 Red Hat Linux 7.1의 Linuxconf편의프로그람을 설명할 때 이미 사용자관리에 대하여 소개하였다. 그러나 Linuxconf는 Linux의 모든 배포물들에서 일반적으로 리용할수 없으므로 Linux 관리를 완전히 정통하기 위해서는 이 도구없이 사용자들을 관리하는 방법을 배워야 한다.

사용자만들기

Linux에서 사용자만들기는 지령userad으로 진행된다(일반적으로 등록부 /usr/sbin에서 찾는다. adduser지령도 역시 같은 방법으로 리용할수 있다.). Red Hat Linux 7.1에 있는 지령useradd는 새로운 사용자회계를 아주 쉽게 만들수 있는 복합도구이다. 가장 간단한 실례로 시작해서 보다 더 복잡한 경우로 넘어 가 보자. 간단한 경우는 모두 기정적인 설정으로 사용자를 만드는 경우이다. 실례로 testuser1라는 새로운 사용자를 만드는 지령은 다음과 같다.

```
# /usr/sbin/useradd testuser1
```

이 지령은 다음과 같은 동작을 수행하여 사용자를 작성한다.

- 통과암호없이 /etc/passwd파일에 사용자에 대한 항목을 만든다. 대부분의 Linux배포물들에서는 사용자가 가입하기전에 통과암호를 요구한다.
- 그 사용자에 대한 사용자ID를 할당한다. Red Hat Linux 7.1에서 기정적인 사용자ID는 다른 사용자들의 ID보다 큰값으로서 500이상이여야 한다.
- 사용자를 해당한 그룹에 추가한다. 일반적으로 Linux에서 이것은 오직 그 사용자만이 속한 그룹을 만든다는것을 의미한다.
- 사용자를 위한 홈등록부(대부분의 Linux에서 /home/testuser1)를 만들고 /etc/skel의 내용을 홈등록부에 복사한다. 골격등록부들은 이 장의 마지막절인 《기정적인 홈등록부들의 설정》에서 더 구체적으로 고찰한다.

주의 Red Hat Linux 7.1에서 새로운 사용자들을 만드는 기정적인 선택항목들은 **/etc/login.defs** 혹은 **/etc/default/useradd**구성파일에서 자동적으로 정의된다.

사용자를 추가하는 useradd지령의 리용은 《통과암호바꾸기》절에서 본 기술을 활용하여 사용자에 통과암호를 지정하는 체계관리기의 자유에 맡긴다.

사용자ID를 할당하는 체계의 기정적인 설정들을 무시하려면 어떻게 해야 하는가? Red Hat Linux 사용자 ID들이 토대하고 있는 개별적인 조직체의 식별번호를 가진 조직을 고찰하자. 이 경우에 -u기발을 리용하여 개별적인 ID에 대한 할당을 요구하게 된다.

```
# useradd -u 10001 testuser1
```

이것은 10001로 지정된 사용자 ID로 testuser1에 대한 회계를 만든다.

마찬가지로 기정그룹으로 사용자에 특별한 그룹을 할당하려면 어떻게 해야겠는가? 실례로 testuser1을 개별적인 그룹에 할당하지 않고 users라는 Red Hat Linux의 기정적인 그룹에 할당하려면 –g기발을 리용할수 있다. 즉

```
# useradd -g users testuser1
```

한단계 더 나가서 testuser1이 users라는 기정그룹이 아니고 그룹1과 그룹2라는 그룹들에 속해 있다고 가정하자. 후에 그룹의 관리에 대한 절에서 보게 되겠지만 정의된대로 사용자를 만들고 그것을 수동으로 그 그룹들에 추가한다. 그러나 useradd는 회계가 만들어 질 때 새로운 사용자를 추가하기 위한 추가적인 그룹들을 정의하도록 –g기발을 제공한다. 즉

```
# useradd –g users G 그룹1,그룹2 testuser1
```

마지막으로 사용자에 대한 대리홈등록부를 지정하기 위하여 -d기발을 리용한다. 즉

```
# useradd –d /other/home/directory testuser1
```

useradd의 기정적인 설정들의 바꾸기

사용자를 작성할 때마다 무시할수 있는 useradd가 리용하는 기정적인 설정들이 몇가지 있다. 실례로 홈등록부들을 모두 /home대신 /users에서 만들수 있다. 또한 모든 새로운 사용자들을 자기들의 고유한 그룹이 아니라 users라는 기정그룹에 속하도록 할수도 있다.

이 기정설정들을 useradd지령의 -D기발과 몇가지 보조기발들을 리용하여 재설정할수 있다. -D기발은 그 지령이 새로운 사용자를 만들지 않지만 새로운 기정설정들을 할당해야 한다는것을 지시한다.

-D기발은 보조기발들과 함께 리용된다. 이제 기정홈등록부경로와 기정그룹들을 각각 재설정하기 위하여 –b와 -g기발을 리용하는 방법을 보기로 하자.

/users로 기정홈등록부경로를 설정하려면 다음의 지령을 리용해야 한다.

```
# useradd  -D –b /users
```

마찬가지로 모든 새로운 사용자들에 대한 기정그룹을 users로 설정하려면 다음의 지령을 리용해야 한다.

```
# useradd  -D -g users
```

이러한 지령들은 또 한개의 지령과 결합될수 있다. 즉

```
# useradd  -D -b /users –g users
```

보조적인 기발들이 없이 useradd –D지령으로 현재 기정값들을 검색할수도 있다.

통과암호들의 바꾸기

통과암호들의 바꾸기는 passwd지령으로 수행된다. 임의의 사용자는 재촉문에 간단히 지령을 입력하여 통과암호를 바꿀수 있다. 그것은 새로운 통과암호에 뒤이어 현재 통과암호를 입구하겠는가 하는 두차례의 확인문을 제시한다. 즉

```
$ passwd
Changing password for test
(current) UNIX password:
New UNIX password
Retype new UNIX password:
passwd: all authentication tokens updated successfully
```

Linux의 많은 판본들에서 passwd지령은 통과암호가 너무 짧지 않는가, 너무 단순하지 않는가, 사용자이름과 너무 비슷하지 않는가, 이미전의 통과암호와 너무 비슷하지 않는가 하는것을 조사하기 위하여 검색한다. 통과암호가 틀리면 다음과 같은 오유통보를 내보낸다.

```
$ passwd
Changing password for test
(current) UNIX password:
New UNIX password:
BAD PASSWORD: it does not contain enough DIFFERENT characters
New UNIX password:
BAD PASSWORD: it is too short
New UNIX password:
BAD PASSWORD: is too simplistic/systematic
passwd: Authentication token manipulation error
```

경고 Red Hat Linux의 선행판본들에서 BAD PASSWORD경고를 무시하면서 사전단어를 통과암호로 리용할수도 있었다. Red Hat Linux 7.1부터는 뿌리사용자가 아니면 더는 그렇게 할수 없다.

뿌리사용자는 passwd프로그람에 인수로 사용자이름을 제공하여 임의의 사용자의 통과암호를 바꿀수 있는 특권을 가진다. 이 경우 재촉문은 오직 새 통과암호만 두번 접수한다.

```
# passwd test
New UNIX password:
Retype new UNIX password:
passwd: a11 authentication tokens updated successfully
```

기정홈등록부들을 설정

이미 《사용자만들기》절에서 골격등록부들에 대하여 잠간 론의하였다. 기정으로 모든 사용자에게는 보통 /home등록부의 보조등록부로서 홈등록부가 주어 진다. 사용자의

회계가 작성될 때 기정파일들의 묶음을 가진 홈등록부가 함께 작성된다. 이 기정파일들의 묶음은 새로운 홈등록부들에 대한 골격등록부를 포함하는 /etc/skel등록부에서 복사된다.

모든 새로운 홈등록부에 어떤 파일을 포함시키기 위하여 그 파일을 간단히 만들어 사용자들의 홈등록부에 있도록 같은 이름을 가진 /etc/skel에 배치한다. 거기에 파일을 배치한 다음 추가된 모든 사용자들은 자기들의 회계가 작성될 때 자기의 홈등록부에서 그 파일을 찾을수 있게 된다.

사용자삭제

사용자삭제는 사용자만들기와 동반되는 처리이다. 지령은 userdel 이다. 이 지령은 useradd지령보다는 간단하다. 사용자회계를 지우고 /etc/passwd와 같은 해당한 체계파일로들로부터 그 내용을 삭제하려면 간단히 사용자이름을 인수로 주면 된다. 즉

 # /usr/sbin/userdel 사용자이름

여기서 문제는 삭제되지 않는 사용자파일이 있는것이다. 동시에 사용자의 홈등록부를 삭제하려면 -r기발을 추가해야 한다. 즉

 # /usr/sbin/userdel -r 사용자이름

이것은 또 한가지 문제를 야기시킨다. 만일 사용자가 삭제에 필요한 파일들을 체계밖에 가지고 있다면 어떻게 해야 하는가? 이것은 user를 삭제한후 find지령을 써서 할수 있다. 이를 위해 사용자를 삭제하기전에 /etc/passwd파일에서 사용자의 ID를 찾아 보고 그다음에 find지령을 주어야 한다. 즉

 # find / -type f -uid 503 -print -exec rm {} \ ;

이 지령의 내용을 해석해 보자.

- 빗선(/)은 맨 웃준위의 등록부로부터 시작하여 전체 등록부구조를 탐색한다는것을 나타낸다.
- -type f는 지령이 파일만을 찾는다는것을 가리킨다.
- -uid 503은 ID 503 을 가진 사용자가 소유한 파일만을 복귀한다는것을 가리킨다.
- -print는 지령실행과정을 추적할수 있도록 탐색된 파일이름을 제시해야 한다는것을 가리킨다.
- exec rm {} \ 은 rm 지령이 탐색된 모든 파일을 완전히 삭제해야 한다는것을 나타낸다.

경고 우에서 언급된 find지령을 리용할때 주의해야 한다. 왜냐하면 체계에 있는 파일들의 소유자(사용자)를 헛갈리거나 지령을 잘못 주는 경우 중요한 자료들을 지워 버릴수 있기때문이다. 사용자가 파일들을 자기의 홈등록부밖에 가지고 있을 때에만 우와 같은 find지령을 리용할수 있다.

그룹관리

련관된 사용자들의 집합체들인 그룹들의 관리는 사용자관리처럼 간단하며 사용자관리처리와 동반된다. Linux는 사용자관리를 비롯하여 그룹들의 작성과 갱신을 자동화하는 지령들을 제공한다.

주의 다른 Linux배포물들에서는 이 지령들이 여기에서 언급한것과는 다르게 동작할수 있다. 따라서 이 책에 포함된 Red Hat Linux 7.1의 복사물을 리용하지 않으면 문서를 탐색해야 한다.

그룹만들기

groupadd지령으로 체계에 새 그룹을 추가할수 있다(이 지령은 일부 배포물들에서는 addgroup으로 쓰인다.). 그룹을 만들기 위해 인수로 그룹이름을 준다. 즉

 # /usr/sbin/groupadd 그룹이름

그룹은 groupadd에 대한 Red Hat Linux 7.1 사용설명서페지로부터 다음의 규칙에 따르는 새로운 사용자번호로 작성되고 할당된다.《기정값은 다른 모든 그룹들보다 더 큰 최소한500이상인 ID값을 리용하게 된다.》 일부 기타Linux배포물들에서는 기정으로 users그룹(100)에 새로운 사용자들을 추가한다.

만일 그룹번호를 정의하려면 번호를 나타내는 -g기발을 리용해야 한다. 즉

 # groupadd -g 503 그룹이름

이것은 etc/group에 다음과 같은 항목을 작성할수 있다.

그룹이름:503:

이 항목은 정의된 이름과 ID를 가진 빈그룹을 보여 준다.

일러두기 만일 그룹을 잘 모르겠다면 현존그룹ID들이 etc/group파일에 있으므로 찾아보아야 한다.

그룹에 사용자들의 추가

공교롭게도 그룹에 사용자들을 쉽게 추가할수 있는 표준적인 프로그람은 없다. 그룹에 사용자들을 추가할수 있는 제일 간단한 방법은 /etc/group파일을 편집하는것이다. 이 파일에서 매행은 그룹에 대한 정의를 나타낸다. 즉

그룹이름:통과암호:그룹id:사용자목록

행의 매 요소들에 대한 의미는 다음과 같다.

- 그룹이름은 그룹에 붙은 이름이다.
- 통과암호는 그룹에 대한 통과암호로서 일반적으로 그룹들에 드문히 적용되고 보통 생략된다.
- 그룹ID는 그룹에 대한 식별번호이며 그 그룹에만 고유한 수값이다.
- 사용자목록은 그룹에 속한 사용자들을 반점으로 구분해서 표시하는 목록마당이다. 실례로 사용자1, 사용자2, 사용자3이 그룹ID가 505이고 그룹이름이 그룹1

이라고 하는 그룹에 모두 속한다고 할 때 행을 다음과 같이 서술할수 있다.

그룹1::505:사용자1,사용자2,사용자3

현재 있는 그룹에 사용자들을 추가하려면 문서편집기로 간단히 /etc/group파일에 해당한 그룹의 사용자목록마당의 끝에서부터 사용자이름들을 반점으로 구분하면서 추가하는 식으로 편집할수 있다.

그룹삭제

그룹삭제는 groupdel지령으로 진행한다. 이 지령은 매우 간단하다. 즉 기발이나 항목들이 필요 없다. 간단하게 인수로 삭제하려는 그룹이름을 주면 그 그룹은 삭제된다. 즉

/usr/sbin/groupdel 그룹이름

그러나 간단한 모든 경우에도 몇가지는 주의해야 한다. 즉

- 그 그룹에 속하고 있는 파일들은 삭제되지 않고 그룹만 변한다.
- 만일 그 그룹이 단일사용자를 위한 초기그룹으로 봉사한다면(다르게 말하면 통파암호파일에 속한 사용자그룹이라면) 그 그룹은 삭제되지 않는다.

첫번째 문제는 사용자파일을 삭제함으로써 사용자가 완전히 삭제되였을 때에만 해결될수 있다. 우선 삭제하고 있는 그룹의 ID에 주의를 돌려야 한다(이것은 /etc/group파일에서 찾아 볼수 있다.). 일단 그룹이 groupdel로 삭제되면 삭제된 그룹에 속한 모든 파일들의 그룹소유권을 바꾸기 위해 find지령을 리용해야 한다. 즉

find / -type f –gid 503 –print -exec chgrp 새그룹이름 () \ ;

이 find지령은 ID가 503인 그룹에 속하는 모든 파일들을 찾고 새 그룹이름으로 된 그룹으로 그 파일들의 그룹소유권을 바꾸기 위해 chgrp지령을 리용한다.

체계기동

많은 Linux사용자들에게 호기심을 자아내는 령역의 하나가 기동순서인데 그때 신비스러운 통보들이 모두 화면우에 잠간 제시된다.

주의 기동통보들은 체계기록파일 /var/log/dmesg에 보관된다. 기록파일들은 이장의 마감에서 고찰한다.

기동과정에 일어 나는 사건

기동과정은 기동시 통보들이 시사하는것보다 실제상 더 단순하다. 기본적으로 기동과정에 두가지 상태들이 있을수 있다.

1. 핵심부기동 이 상태에서는 핵심부가 기억기에 적재되여 매 장치구동기를 초기화하면서 통보문들을 내보낸다.
2. init프로그람실행 핵심부가 적재되여 장치초기화를 끝낸후 init프로그람이 실행된다. init는 본질적인 체계데몬들과 기동시에 적재되도록 정의된 기타 쏘프트웨어를 비롯하여 모든 프로그람들의 설치를 조종한다.

init프로그람

이 절에서는 init프로그람에 대해서 더 구체적으로 고찰한다. 왜냐하면 여기에서 기동과정에 설치할 프로그람들을 쉽게 선택할수 있기때문이다. Init는 새로운 처리들을 시작하다가 그것들을 끝내면 다른 처리들을 다시 시작하는 일감을 가진다. 이에 대한 완전한 실례로 Linux에서 가상적인 가입조종탁(login console)들을 제공하는 처리들을 들수 있다. 대부분 Linux체계들에서는 기동시에 초보적으로 6개의 조종탁들이 설치된다. 어느한 조종탁창문에서 탈퇴하면 그 처리는 없어 지고 init는 즉시 새로운 조종탁설치처리를 시작하므로 항상 6개의 조종창문들이 유지된다.

init프로그람의 조작을 관리하는 규칙들은 파일 /etc/inittab에 보관된다. Red Hat Linux 7.1의 기정적인 /etc/inittab파일은 다음과 같다.

```
#
# inittab       This file describes how the INIT process should set up
#               the system in a certain run-level.
#
# Author:       Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#              Modified for RHS Linux by Marc Ewing and Donnie Barnes
#

# Default runlevel. The runlevels used by RHS are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have
#   networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - XII
#   6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:

# System initialization.
Si: :sysinit:/etc/rc.d/rc.sysinit

10:0 wait:/etc/rc.d/rc 0
11:1 wait:/etc/rc.d/rc 1
12:2 wait:/etc/rc.d/rc 2
13:3 wait:/etc/rc.d/rc 3
14:4 wait:/etc/rc.d/rc 4
15:5 wait:/etc/rc.d/rc 5
16:6 wait:/etc/rc.d/rc 6

# Things to run in every runlevel
ud::once:/sbin/update
```

```
# Trap CTRL-ALT-DELETE
ca::ctr1a1tde1:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left. Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfai1:/sbin/shutdown -f -h +2 "Power Failure; System Shutting
   Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown
   Cancelled"

# Run gettys in standard runlevels
1 2345 respawn /sbin/mingetty ttyl
2 2345 respawn /sbin/mingetty tty2
3 2345 respawn /sbin/mingetty tty3
4 2345 respawn /sbin/mingetty tty4
5 2345 respawn /sbin/mingetty tty5
6 2345 respawn /sbin/mingetty tty6

# Run xdm in runlevel 5
# xdm is now a separate service
x:5:respawn:/usr/bin/XII/xdm -nodaemon
```

inittab파일을 작성하는 방법을 배우는것은 중요치 않지만 이 파일의 내용을 파악하는것은 중요하다.

Linux는 여러가지 실행준위들중 어느 한가지 준위의 체계를 가진다. 실행준위는 체계의 현재 상태를 식별하는 값인 동시에 init가 그 체계상태에서 실행하고 유지하는 처리들을 식별하는 값이다. inittab파일의 첫번째 내용인 id:3:initdefault는 기동시에 설치하는 기정실행준위를 정의한다. 앞의 실례에서 그것은 다중사용자조종탁방식이며 실행준위 3이다. inittab파일에서 그다음에 있는 매 항목은 그 항목의 두번째 마당에서 적용하는 실행준위를 정의한다(매 마당은 두점으로 구분된다). 따라서 실행준위3에 다음의 행들이 해당된다.

```
l3:3:wait:/etc/rc.d/rc 3
1:2345:respawn /sbin/mingetty ttyl
2:2345:respawn /sbin/mingetty tty2
3:2345:respawn /sbin/mingetty tty3
4:2345:respawn /sbin/mingetty tty4
5:2345:respawn /sbin/mingetty tty5
6:2345:respawn /sbin/mingetty tty6
```

첫번째 행은 시동스크립트/etc/rc.d/rc3을 실행한다. 이것은 등록부 /etc/rc.d

/rc3.d에 있는 모든 스크립트들을 기동시킨다. 이 스크립트들은 sendmail, PCMCIA봉사, 인쇄기데몬, crond와 같은 체계초기화를 시작하는데 필요한 프로그람을 서술한다. 일반적으로 이 스크립트들을 편집하거나 갱신하려고 하지는 않을것이다. 즉 그것들은 체계기정설정들이다. 마지막 6행은 Linux에 제공되는 6개의 가상조종탁들을 설정한다.

주의 실행준위들과 그 준위스크립트들은 Linux배포물에 따라 크게 변한다. 다중사용자조종탁방식에는 3준위가 아닌 준위도 있을수 있으며 다중사용자조종탁방식스크립트가 있는 등록부도 달라 질수 있다.

기본주목을 돌려야 할것은 rc3.d등록부에서 실행되는 마지막 스크립트가 S99local일것이라는것이다. 그 스크립트는 실제적으로 파일etc/rc.d/rc.local에로의 련결이다. 즉 여기에 기동시에 설치하고 싶은 기동프로그람들을 선택하여 놓을수 있다.

다른 Linux체계들에서 /etc/inittab 구조는 /etc/rc.d등록부환경처럼 달라 질수 있다. 그러나 모든 체계들에서 파일 /etc/rc.d/rc.local은 기동지령들을 추가할수 있는 파일로 표현될것이다.

rc.local파일의 리용

Red Hat Linux 7.1의 기정인 rc.local파일은 가입재촉문을 설정하며 아무것도 하지 않는다. 즉

```
#!/bin/sh

# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

if [ -f /etc/redhat-release ]; then
R=$(cat /etc/redhat-release)
arch=$(uname -m)
a="a"
case "_$arch" in
      _a*) a="an";;
_i*) a="an";;
esac

NUMPROC='egrep -c "^CPU [0-9] +" /proc/stat'
if [ "$NUMPROC" -gt "1" ]; then
      SMP="$NUMPROC-processor "
if [ "$NUMPROC" = "8" -o "$NUMPROC" = "11" ]; then
              a="an"
else
a=" a"
fi
```

```
fi
# This will overwrite /etc/issue at every boot. So, make any
# changes you
# want to make to /etc/issue here or you will lose them when you
# reboot.
echo "" > /etc/issue
echo "$R" 》 /etc/issue
echo "핵심부 $(uname -r) on $a $SMP$(uname -m)" 》 /etc/issue

cp -f /etc/issue /etc/issue.net
echo 》 /etc/issue
fi

 touch /var/lock/subsys/local
```

이것은 복잡해 보이지만 아주 간단하다. 그 단계들은 다음과 같다.

1. 리용되고 있는 Red Hat Linux를 식별한다.
2. 장치방식(Intel계의 i386 이나 Alpha계의 axp와 같은)을 식별한다.
3. 콤퓨터의 처리소자 또는 CPU수를 식별한다.
4. /etc/issue에 가입재촉문의 내용을 보관한다.

rc.local파일이 표준적인 쉘스크립트이므로 환경변수들의 할당과 프로그람들의 설치를 비롯하여 쉘스크립트에 부합되는 작업들을 여기에서 진행할수 있다. 실례로 자기가 관리하는 체계에서 아래와 같은 지령으로 rc.local파일에 자료기지데몬을 설치할수 있다.

```
# /usr/local/Minerva/bin/msqld &
```

자기가 실행시키려는 기동프로그람을 임의로 추가하기 위하여 rc.local파일을 자유롭게 리용할수 있다. 실례로 망에 있는 개별적인 콤퓨터가 기동을 끝낼 때마다 전자우편을 반입하려면 I have booted –콤퓨터이름이라는 대상행을 가진 사용자이름@*some.domain*에게 고속우편통보를 보내기 위해서 다음과 같은 지령을 실행하도록 문의하는 콤퓨터에서 rc.local파일을 리용할수 있다. 즉

```
# /bin/mail -s "나는 기동하였다 – XX콤퓨터" X사용자@some.domain
```

완료(Shutting Down)

체계시동과 밀접히 관계된것이 체계완료이다. 모든 다중과제조작체계들에서와 마찬가지로 Linux에서도 하드디스크구동기들에 기억된 자료의 손실을 막기 위해 체계를 깨끗이 완료하는것이 필수적이다. 일반적으로 이것은 뿌리사용자가 shutdown지령을 실행함으로써 수행된다. 즉

```
# shutdown –h now
```

now 인수는 즉시 체계를 완료하기 위한 프로그람을 의미한다. –h기발은 완료후에 체계가 정지된다는것을 의미한다. System Halted통보문이 제시되는데 이것은 콤퓨터전원을 꺼도 된다는것을 알리는것이다.

shutdown지령과 밀접한 련관을 가지는것이 reboot지령이다. 그것은 깨끗이 완료된 체계를 다시 기동한다. 앞서 언급된 /etc/inittab파일에서 Ctral+Alt+Delete건조합을 무효로 하지 않는 한 이 건조합을 눌러 reboot지령을 실현할수도 있다.

일러두기 /etc/inittab파일에서 Ctrl+Alt+Delete항목을 무효화할수 있다는것은 아주 좋은 사상이다. 기정으로 Red Hat Linux 7.1은 모든 사용자들이 이 건조합을 리용하여 재기동할수 있도록 한다. Linux콤퓨터를 호출하고 있는 매 사용자의 작업을 새치기하기 위해 그 누구의 실패도 바라지 않는다.

crond에 의한 일감의 일정작성

Linux와 같은 다중사용자봉사기급조작체계의 큰 장점의 하나가 많은 일감들이 개별적인 방식으로 발생할수 있다는것이다. 만일 우편봉사기, Web봉사기 또는 FTP봉사기로서 Linux를 리용하면 그 응용프로그람들중에서 어느 하나가 봉사에 따르는 결합들과 요구들에 응답할 때마다 개별적인 동작이 발생한다. 또한 개별적인 사용자들과 체계관리기는 개별적인 방식으로 이전에 명세화된 과제들을 수행하기 위하여 Linux를 구성할수 있다.

이 동작은 crond라는 데몬의 기능으로 실행된다. 이 데몬은 표준이며 보통 체계가 기동할 때 설치된다.

crond는 어떻게 작업하는가?

crond가 작업하는 방법은 비교적 간단하다. crond가 기동하면(일반적으로 기동시에) 그것은 1분 간격으로 동작하면서 1분동안에 어떤 일감들이 실행되도록 명세화되였는가를 검사한다. 그후 일감은 실행되고 출구결과는 일감을 명세화한 사용자에게 전자우편으로 전송된다.

crond가 1분 간격으로 자기의 구성파일에 등록된 자료검인들을 검사하므로 crond처리를 재기동하지 않아도 명세화에 대한 임의의 변경들을 다 주목하게 된다.

일감의 일정작성

일감의 일정작성은 쉬운 문제이다. 일정작성된 모든 일감들은 그 사용자에 대한 개별적인 구성파일(crontab라는 파일)에 보관된다. 그 파일의 매행은 일정작성된 일감을 서술한다. 일정작성된 일감들을 만드는 crontab파일을 작성하는 방법을 보기전에 먼저 crontab파일내용의 형식을 보기로 하자.

매 항목은 다음과 같은 형식을 취한다.

  시간-날자 지령

시간-날자항목은 5개의 수자마당으로 이루어 지며 매 수자는 공백으로 구별되는데 그것은 일감이 실행되야 할 시간을 가리킨다. 5개의 마당은 다음과 같은 순서로 되여있다. 즉

- 분: 0~59사이의 값을 가진다.
- 시간: 0~23사이의 값을 가진다.
- 날자: 0~31사이의 값을 가진다.
- 달: 0~12사이의 값을 가진다.
- 요일: 0~7사이의 값을 가지는데 0과 7은 둘다 일요일을 나타낸다.

이러한 모든 마당에 몇가지 규칙은 유연성을 제공한다. 즉

- 수자범위들이 리용될수 있다. 실례로 시간마당에서 1~3은 오전 1시, 2시, 3시에 그 지령을 명세화한다는것을 의미한다. 또한 요일마당에서 2~4는 화요일, 문자들은 리용가능한 마당에서 수자에 대신에 쓸수 있는 문자이다. 수요일, 목요일에 일감을 명세화한다.
- 범위는 하나이상의 증분값으로 단계화될수 있다. 실례로 저녁12시이외의 매 시간마다 지령을 실행하기 위해 범위 0~23을 빗선으로 구분하여 2단계로 결합한다.
- 별표(*)는 가장 작은값~가장 큰값까지 마당에 해당한 입구령역을 지적한다. 따라서 날자마당을 별표로 하면 0~31과 같고 요일마당도 0~7로 된다.
- 달이나 날자이름의 첫 3개

표본시간과 날자

시간-날자마당에 대한 표본들을 몇가지 보자.

0 1 * * *	이 마당은 매일 오전 1시에 일감을 실행하여야 한다는것을 보여준다.
30 14 * * 1	이 마당은 매주 월요일 오후 2시반에 일감을 실행하여야 한다는것을 보여 준다.
0 12 1 * *	이 마당은 매달 초하루 낮 12시에 일감을 실행하여야 한다는것을 보여 준다.
0 12 * 1 mon	이 마당은 매해 1월 월요일 낮 12시에 일감을 실행하여야 한다는것을 보여 준다.
0 12 2 feb *	이 마당은 매해 2월 2일 낮 12시에 일감을 실행하여야 한다는것을 보여 준다.

지령마당

시간-날자마당은 지령마당과 한개이상의 공백으로 분리되여 있으며 그 행의 마감까지 실행한다. 지령들은 /bin/sh 쉘에 의하여 처리된다.

실례로 crontab항목

0 1 * * * /usr/local/bin/backup

은 매일 오전 1시에 프로그람 /usr/local/bin/backup을 실행하게 한다.

때때로 지령(우편지령과 같은)들이 표준입력을 통하여 입력되는 정보를 요구할수 있다. 이 정보는 퍼센트기호(%)로 표시된다. 첫번째 퍼센트기호는 표준입력의 시작을 나타내며 두번째부터 나타나는 퍼센트기호들은 표준입력에서 새행시작기호로 표시된다.

따라서 crontab항목

30 14 * * fri /bin/mail -s “TGIF” armand@landegg.edu%Thank God It’s Friday%%Me

은 금요일 오후 2시30분에 armand@landegg.edu에 다음의 전자우편통보를 전송한다.

Thank God It’s Friday
Me

crontab파일의 편집

crontab -e지령으로 crontab파일을 편집할수 있다. 두가지 가능성이 있다. 한가지 방법은 crontab파일에서 서술되는 모든 항목들을 포함하는 파일을 작성하고 crontab지령으로 그것을 적재하는것이다. 다른 방법은 직접 crontab -e지령으로 crontab파일을 편집하는것이다.

파일로부터 적재 파일로부터 항목들을 적재하려면 우선 crontab파일에 나타나게 하려는 모든 항목들을 포함하는 파일을 작성해야 한다. 아래에 두개의 내용을 포함하는 한개의 표본파일을 보여 주었다.

```
0  1 * * *      /usr/local/bin/backup
30 14 * * fri   /bin/mail -s "TGIF" armand@1andegg.edu%Thank God It's Friday%%Me.
```

이 파일은 cronjobs와 같은 알맞는 이름으로 보관해야 한다.

일단 파일이 작성되고 보관되면 그것은 다음과 같은 지령을 실행하여 사용자의 crontab파일에 적재할수 있다.

```
$ crontab cronjobs
```

cronjobs의 항목들은 사용자의 crontab파일에 현재 내용들을 덧쓰기한다. 이 방법을 리용하여 임의의 사용자는 자기의 crontab파일을 조종할수 있다.

뿌리사용자는 임의의 사용자에 대한 crontab항목들을 편집하도록 하는 특권을 가진다. -u기발을 리용하면 뿌리사용자는 뿌리사용자의 crontab파일을 대신하는 다른 사용자의 crontab파일을 정의할수 있다. 실례로 지령

```
# crontab –u username cronjobs
```

는 사용자 username의 crontab파일과 같은cronjobs파일을 적재한다.

crontab파일을 직접 편집 파일을 작성하고 그것을crontab파일에 적재하는 방법대신에 crontab지령은 직접 사용자가 crontab파일을 편집하도록 -e기발을 제공한다.

기정으로 crontab –e는 vi편집기를 리용하여 crontab파일을 편집하도록 한다. vi편집기는 강력하지만 일반적으로는 hard-to-master편집기가 Unix사용자들속에서 오래동안 쓰이였다. xedit같은 다른 편집기를 리용하려면 이 프로그람에 EDITOR환경변수의 값을 설정해야 한다. 즉

```
$ export EDITOR=xedit
```

그러면 지령

```
$ crontab -e
```

은 그림 17-1과 같이 지정된 편집기를 리용하여 편집하려는 crontab파일을 연다.

파일을 crontab파일에 적재할 때와 같이 -u기발을 리용하여 뿌리사용자는 직접 다른 사용자의 crontab파일을 편집할수 있다. 즉

```
$ crontab –u username –e
```

***crontab*파일의 내용보기** crontab파일내용을 보려면 간단히 -l기발을 리용해야 한다. 즉

```
$ crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
```

```
# (/tmp/crontab.555 installed on Mon Jul 13 00:07:05 2001)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp  $)
0 1 * * *       /usr/local/bin/backup
30 14 * * fri   /bin/mail -s "TGIF" armand@1andegg.edu%Thank God It's
Friday%%Me
```

그림 17-1. xedit로 crontab파일을 편집

crontab파일은 하쉬기호(#)로 시작하는 몇개의 설명행을 가진다는것을 명심해야 한다.

crontab파일편집에서처럼 뿌리사용자는 -u기발로 임의의 사용자의 crontab파일내용을 볼수 있다. 즉

crontab –u 사용자이름 -l

***crontab*파일의 삭제** 사용자의 crontab파일내용을 삭제하려면 -r기발을 리용해야 한다. 즉

crontab –r

또한 뿌리사용자는 -u기발로 사용자의 crontab파일을 지울수 있다.

crontab –u 사용자이름 –r

기록관리

Linux와 모든Unix 계들의 한가지 우점은 체계에서 실행하는 각종 데몬들과 프로그람의 동작들을 기록하는 표준화된 기구들을 제공하는것이다. 이 사건기록들은 보호위반동작으로부터 하드웨어오유경고에 이르기까지 모든 사건들을 담고 있으면서 체계의 사용과정은 물론 체계에서 제기되는 문제들을 수정하는데 리용될수 있다.

실례로 아래에 인용한 Red Hat Linux 7.1체계의 기본기록파일(/var/log/messages)은 시기마다 제기된 사건들에 대한 각이한 많은 정보들을 제공한다.

```
May  9 11:32:58 laptop71 network: Shutting down device eth0: succeeded
May  9 11:33:00 laptop71 network: Bringing up device eth0: succeeded
May  9 11:33:00 laptop71 kernel: eth0: flipped to 10baseT
May  9 11:33:01 laptop71 xinetd[3511]: Exiting...
May  9 11:33:02 laptop71 xinetd: xinetd shutdown succeeded
May  9 11:33:04 laptop71 xinetd[5598]: chargen disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: ftp disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: time disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: time disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: telnet disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: talk disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: rsync disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: shell disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: login disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: exec disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: ntalk disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: linuxconf disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: finger disabled. removing
May  9 11:33:04 laptop71 xinetd[5598]: echo-udp disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: echo disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: daytime-udp disabled. removing
May  9 11:33:04 laptop71 xinetd[5598]: daytime disabled removing
May  9 11:33:04 laptop71 xinetd[5598]: chargen-udp disabled, removing
May  9 11:33:04 laptop71 xinetd[5598]: xinetd Version 2.1.8.9pre14
     started with
May  9 11:33:04 laptop71 xinetd[5598]: libwrap
May  9 11:33:04 laptop71 xinetd[5598]: options compiled in.
May  9 11:33:04 laptop71 xinetd[5598]: Started working: 0 available
     service
May  9 11:33:05 laptop71 xinetd: xinetd startup succeeded
May  9 11:33:21 laptop71 kernel: eth0: flipped to 10baseT
May  9 11:33:38 laptop71 kernel: smb_trans2_request: result=-104,
     setting invalid
May  9 11:33:39 laptop71 kernel: smb_retry: successful, new pid=2190,
     generation=8
```

여기서 실지 배우는것은 무엇인가? 망프로그람은 망대면부 eth0이 10baseT형망으로 완료되고 재기동되였으며 xinetd가 재기동하기전에 각종 봉사들을 취소하였고 2190의 PID로 전송하는 한개의 smb(Samba)파일이 있었다는것을 통지한다.

무엇을 기록하는가?

Linux체계에서 여러가지 형태의 사건기록들을 구별하는것이 중요하다. 기본적으로 두가지 형태의 사건기록 즉 체계기록(사건기록)들과 응용프로그람기록(사건기록)들이 있

다. 이 절에서는 모든 체계들이 가지고 있는 체계기록들을 고찰한다. 응용프로그람전용기록들은 실행하고 있는 응용프로그람들에 따라 결정되며 기록들을 생성하는 이 응용프로그람들을 구성하는 방법에 따라 좌우된다.

체계기록에서는 적재된 모듈의 정보를 포함하는 핵심부의 통보와 경고문,체계에서 처리된 통보의 흔적을 제공하는 sendmail데몬(daemon)에서 제시한 자료 그리고 인증(가입)시험의 성공과 실패에 대한 통보문을 찾아 볼수 있다.

체계기록은 기동시에 적재하는 sys logd데몬(daemon)으로 만든다. 데몬은 핵심부, 우편체계, syslogd리용을 위해 구성된 사용자프로그람 그리고 login프로그람을 비롯하여 인증프로그람과 같은 각종 체계처리로부터 엄격한 8가지 준위들로 통보문을 호출한다.

이 통보문들의 준위는 다음과 같은 엄격한 순서를 유지한다.

- debug
- info
- notice
- warning
- crit
- alert
- emerg

이 준위들은 여러가지 형태의 정보에 대한 기록을 작성하도록 syslogd에 지시하는 /etc/syslog.conf파일에서 리용한다. 이 파일은 한개이상의 공백으로 구별되는 2개 마당 즉 기능준위목록과 기록파일위치를 포함하는 다중항목을 가지고 있다.

기능준위목록은 반두점으로 구분된 기능준위조들의 목록이다. 기능들은 mail,kern(핵심부를 위한), user(사용자프로그람에 대한), auth(인증프로그람에 대한)와 같은 이름을 가진 기능으로 표현된다. 표본기능준위조에는 다음과 같은것이 포함된다.

- mail.err: 우편데몬에 의해서 발생된 오유
- *.info: 모든 정보통보문
- kern.emerg: 핵심부로부터 발생한 비상사건

이것이 어떻게 작업하는가를 알기 위해Red Hat Linux 7.1에 있는 기정 /etc/syslog.conf파일을 보기로 하자.

```
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                                  /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;news.none;authpriv.none                /var/log/messages
# The authpriv file has restricted access.
authpriv.*                                              /var/log/secure

# Log all the mail messages in one place.
mail.*                                                   /var/log/maillog

# Log cron stuff
```

```
cron.*                                                  /var/log/cron

# Everybody gets emergency messages, plus log them on another
# machine.
*.emerg                                                  *

# Save mail and news errors of level err and higher in a
# special file.
uucp,news.crit                                          /var/log/spooler

# Save boot messages also to boot.log
local17.*                                               /var/log/boot.log
```

첫번째 중요한 행은

```
*.info;mail.none;news.none;authpriv.none               /var/log/messages
```

이다. 이 행은 우편, 새소식, 인증(mail.none;news.none;authpriv.none)외의 모든 기능으로 발생하는 정보통지문을 /var/log/messages파일에 기록한다. 이것은

```
authpriv.*                                              /var/log/secure
```

에 따르며 모든 인증통보문을 /var/log/secure에 배치한다.

다음행은 /var/log/maillog에 배치되야 할 모든 우편기록통보를 정의한다.

```
mail.*                                                  /var/log/maillog
```

이 뒤로

```
uucp,news,crit                                          /var/log/spooler
```

가 따르는데 이것은 우편 및 새소식과 관련된 통보들을 /var/log/spooler에 기록한다.

주의할 점은 첫째로 기록통보들이 각이한 파일들로 갈라 진다는것이다. 여기서 중요한것은 매 기록파일마다 크기를 조종하기 쉽게 련관된 정보를 보존함으로써 기록통보들을 내리추적하기가 쉽다는것이다. 만일 매 기능에 따르는 매 준위의 통보문이 한개의 단일한 기록파일에서 끝난다면 그 정보는 사실상 쓰이지 못하므로 그 파일의 정보량은 조종하기 힘들다.

/etc/syslog.conf를 편집하여 사건기록전략을 변화시키려면 syslog.conf파일을 편집하여 다음의 지령으로 구성을 다시 하도록 syslogd 에 지시할수 있다.

```
# kill -HUP `cat /var/run/syslogd.pid`
```

거꿀인용부호를 주의해서 써야 한다. 이 기호에 들어 있는것은 kill -HUP지령에 대한 첨수로서 지령이 실행되여 결과표준출력으로 제공되여야 한다는것을 보여 준다. kill지령의 -HUP기발은 그 처리가 구성파일을 다시 읽지만 실행을 유지해야 한다는것을 가리킨다.

기록의 순환

기록을 실용적인것으로 되게 하려면 규칙적인 기준에 따라서 순환시켜야 한다. 이것은 기록들의 크기를 관리하기 쉽게 하며 더 이상 불필요한 낡은 정보에 따르는 기록들을 삭제한다. 또한 기록들의 크기가 계속 커지므로 그것들을 순환시켜 그것들이 차지해서리용하고 있는 디스크공간을 개방시킨다.

기록순환에 대하여 가장 간단한 전략은 그것들을 삭제하고 syslogd을 재기동하는것이다. syslogd를 재기동할 때 새것을 창조하며 삭제된것을 재배치하기 위해서 기록파일을 비운다. 실례로 지령

```
# rm /var/log/messages
# kill -HUP 'cat /var/run/syslogd.pid'
```

은 /var/log/messages파일을 삭제하고 빈 새로운 /var/log/messages파일을 만들기 위하여 syslogd를 재기동시킨다.

이러한 파일삭제방법은 전통적인 기록들이 각종 목적에 봉사할수 없는 단일사용자나 홈체계에서 널리 리용된다. 그러나 다중사용자봉사기에서 전통적인 정보는 특히 보호위반의 추적에 대해서 특별한 값을 가진다. 이 경우 전략은 좀 다르다. 실례로 전통적인 기록중 한가지를 작성하려면 현행 기록을 새파일이름으로 옮기고 다음에 새로운 현행 기록파일을 작성하기 위해 syslogd 를 재기동시켜야 한다. 즉

```
# mv /var/log/messages /var/log/messages.1
# kill –HUP 'cat /var/run/syslogd.pid'
```

또한 전통적인 기록파일중 두개 파일을 작성하려면 두번째 생성파일이름으로 첫번째 생성파일을 옮기고 다음에 첫번째 생성파일이름으로 현행 기록을 옮겨야 한다. 즉

```
# mv /var/log/messages.1 /var/log/messages.2
# mv /var/log/messages /var/log/messages.1
# kill –HUP 'cat /var/run/syslogd.pid'
```

대부분체계에서 이 처리를 자동화하고 매주 설정된 시간에 그것을 실행하기를 원할것이다. 그러자면 우선 기록파일들을 순환하는데 필요한 동작들을 수행하는 스크립트를 만들어야 한다. 실례로 기록중 한개 기록을 만드는 Linux봉사기에서 다음과 같은 스크립트을 작성할수 있다.

```
#!/bin/sh

mv /var/log/messages /var/log/messages.1
mv /var/log/secure /var/log/secure.1
mv /var/log/maillog /var/log/maillog.1
mv /var/log/spooler /var/log/spooler.1
kill - HUP  'cat /car/run/syslogd.pid'
```

국부(/usr/local/bin/newlogs와 같은)에 있는 본문편집기로 스크립트파일을 작성한 다음 실행파일로 만든다. 즉

```
# chmod 755 /usr/local/bin/newlogs
```

다음에 이 장의 앞에서 언급한 방법들을 리용하여 뿌리사용자의 crontab파일을 편집하고 적당한 내용을 추가하는것이 필요하다. 실례로 매주 일요일 오전 12시 01분마다 그 스크립트을 실행하려면 다음과 같은 항목을 줄수 있다.

```
1 12 * * sun  /usr/local/bin/newlogs
```

일러두기 Red Hat Linux 7.1은 /etc/logrotate.conf파일에 기초하여 기록순환처리를 자동화한다. 이 파일의 첫 두개지령은 4주분의 회복기록을 가지고 매주마다 기록을 순환시킨다. 실례로 /var/log/messages.4를 통하여 /var/log/messages와 /var/log/messages.1과 같이 5주분의 체계기록을 가질수 있다.

▶미리 알아두기

이 장에서는 사용자들과 관계되는 체계관리과제들 즉 명세화 및 기록들에 대하여 고찰하였다.

제18장《주변장치의 리용》에서는 대부분의 일반용 PC주변장치 즉 인쇄기와 모뎀을 구성하고 리용하는 방법을 고찰한다.

여기서는 PostScript와 PCL 인쇄기를 구성하고 Linux/Unix인쇄체계기구들을 찾는 방법을 배우게 된다.

제18장에서는 또한 모뎀구성을 설명하고 모뎀을 검사하고 리용하는 minicom프로그람을 시험해 본다.

제18장. 주변장치의 리용

이 장에서는 가장 일반적인 류형의 주변장치들인 인쇄기와 모뎀들의 구성방법을 고찰한다. 우선 Linux포구들의 동작방식을 리해하고 Linux인쇄체계들을 조종하는 파일들을 본 다음 모뎀을 호출하는데 쓸수 있는 Linux표준말단모방도구인 minicom을 정통하게 된다. 또한 USB마우스와 건반을 구성하는데 필요한 각이한 체계들을 시험하게 된다. 그러나 Linux장치구성기능이 어떻게 동작하는가를 리해하자면 우선 Linux가 PnP(Plug and Play)로 동작하는 방법에 대한 기본적인 리해를 가져야 한다.

Linux와 PnP

Linux는 실제로 PnP조작체계가 아니다. 그러나 콤퓨터에 접속된 하드웨어를 검출할수 있는 프로그람들을 가지고 있다. 이러한 모순점은 약간한 설명을 요구하고 있다.

PnP는 콤퓨터에 접속한 장치들이 검출되고 콤퓨터내부의 각종 통신통로에 자동적으로 구성되는 체계이다. PnP체계는 4가지 구성요소 즉 PnP Bios, PnP주기판, PnP장치, PnP연산체계로 이루어 진다.

현재 대부분의 BIOS들과 장치들은 PnP이다. PnP주기판이 있다고 가정하면 BIOS는 구동기들과 건반 그리고 마우스와 같은 장치들에 대한 통로와 포구, 주소를 설정한다. 이 구성은 조작체계가 적재되기전에 완성한다. Linux는 여러가지 성공적인 조작으로 하드웨어를 검출하기 위하여 미리 설정한 통로들을 리용할수 있다. Linux검출의 성과는 문제의 장치형에 의존한다.

통로, 주소, 포구

PnP는 장치들과 CPU와 RAM과 같은 기본구성요소들사이의 통신을 설정한다. 콤퓨터장치들은 각종 형태의 통로나 주소 혹은 포구로 통신한다.

IRQ 장치들은 CPU로부터 봉사를 받기 위하여 IRQ나 새치기요구를 리용할수 있다.

I/Q 공간은 콤퓨터의 다른 부분사이 정보교환을 위해 RAM에 보류된다. 이 공간은 I/O나 입출구주소로 설계된다.

DMA 일부 장치들은 DMA나 직접기억호출 혹은 통로를 우회할수 있다.

각이한 형태의 장치들은 콤퓨터에 접속하는 방법에 따라 각이한 통로와 주소혹은 포구를 요구한다. 만일 모뎀이나 음성기판과 같은 장치에 문제가 제기되면 Linux는 틀린 IRQ 포구나 PnP주소 혹은 DMA통로에서 이 장치를 찾을수 있다.

ISA

많은 주변장치들은 ISA(Industry Standard Architecture)방식으로 제작된 이전시기의 콤퓨터들에 설치되여 있다.

Plug-and-Play ISA는 1993년에 도입되였으며 대부분 ISA장치들은 PnP표준들에 적합하다. 공교롭게도 PnP ISA장치들은 자주 조작체계가 기대하는 통로를 얻지 못하므로 통로나 포구 혹은 주소를 수동으로 할당해야 한다.

이것은 망기판에서 어느정도 공통적인 문제로 된다. 망기판의 수동적인 구성은 제28장 《이써네트망에서 Led Hat Linux 7.1의 구성》에서 더 구체적으로 고찰한다.

PCI

PCI(Peripheral Component Interconnect)모선은 ISA장치의 결함을 퇴치하여 개발했다. PCI모선은 더 빠르고 일부 IRQ통로들을 공유할수 있다. 일부 PnP BIOS 차림표를 리용하여 개별적인 PCI장치에 직접 IRQ 와 I/O 및 DMA통로를 할당할수 있다.

주의 가속도형기판(Accelerated Graphics Card)들은 비데오기판과 CPU사이에서 도형을 조종하기 위해 제작된 전형적인 PCI기판이다.

External

External장치들은 보통 구성하기가 더 쉬운데 그것은 그것들이 개별적인 통로들을 가지고 있지 않기때문이다. 모뎀이나 인쇄기와 같은 External장치들은 일반적으로 전용적인 물리적포구에 접속된다. 일반적으로 직렬 또는 병렬결합인 이 포구에는 이미 통로들이 할당되여 있다. 접속된 장치들은 같은 통로들을 리용한다.

External장치에서 이 원리에 례외로 되는것이 USB이다.

USB

USB(Universal Serial Bus)는 정확히 말해서 콤퓨터우의 external련결이 아니다. 모든 USB뿌리하브(root hub)는 리론적으로는 127개까지의 주변장치들을 조종할수 있다. USB는 PnP와 콤퓨터가 실행하는 기간 장치들을 추가하고 삭제하는 《강렬한 교환(hot swapping》을 지원한다. 조작체계는 자동적으로 새로운 구성을 인식하도록 지원된다.

불행하게도 Linux는 현재 모든 USB장치를 다는 제공하지 못한다. Red Hat Linux 7.1은 광범한 USB장치들을 제공하지만 USB망기판들을 지원하지 못한다.

그것은 강력하지 못한 USB하브로 작업하는것은 드물고 USB플로피구동기들로부터의 기동을 지원하지 못한다. 하지만 USB에 대한 Linux지원을 개선하기 위하여 끊임 없이 연구를 진행하고 있는 개발자들이 많다. 최소한의 정보에 대해서는 Linux USB Web싸이트 http://www.linux-usb.org를 참고할수 있다.

Red Hat Linux 7.1로 기동하면 강렬한 접속(hot plugging)이 지원된다. 그러나

Linux는 요구되는 장치들이 /lib/modules/2.4.2-2자료기지에 없으면 접속(plug) 혹은 설치를 인정하지 않는다. 더 구체적인 정보를 알려면 Linux Hotplugging Web싸이트 http://linux-hotplug.sourceforge.net를 참고할수 있다.

경고 이에 대한 Linux USB문서는 완성된것이 없다. 따라서 Linux USB보조체계 안내서(http://www.linux-usb.org/USB-guide/book1.html)의 첫페지에 언급된것처럼 《일부 〈현재 Linux USB문서〉는 완전한 추측이며 특히 후에 알려지고 더 확장된 장치들에 대하여서는 더하다.》

인쇄기

제15장 《체계구성을 위한 Linuxconf와 기타도구들의 리용》에서 이미 Printconf편의프로그람을 리용하여 Red Hat Linux 7.1에서 인쇄기들을 쉽게 구성하는 방법을 보았다. 다른 두가지 선택항목들이 있다. 이 장에서는 수동으로 혹은 Apsfilter 체계의 방조를 받아서 인쇄기들을 구성하는 기초들을 고찰하게 된다. 이 프로쎄스에 대해서는 http://www.linuxdoc.org/HOWTO/Printing-HOWTO /index.html 에 있는 Printing HOWTO를 참고할수 있다.

어느 인쇄기를 리용하겠는가?

Linux에서 인쇄기구성을 보기전에 Linux에서 어느 인쇄기로 작업하며 그것들을 어떻게 리용하려고 하는가를 먼저 고찰하자. 언급된것을 제외하고 대부분의 인쇄기는 다음과 같은것으로 지원된다.

PostScript 인쇄기 대부분의 Linux쏘프트웨어는 인쇄과정에 PostScript 출력을 생성하므로 PostScript인쇄기를 선택하는것이 아주 좋다. 물론 급이 낮은 레이자인쇄기들과 잉크분사식인쇄기들은 PostScript 를 지원하지 못한다.

Ghostscript가 지원하는 PostScript형이 아닌 인쇄기 자기의 인쇄기가 PostScript를 지원하지 못하지만 Printer Control Language(PCL)를 지원하면 그 인쇄기는 쏘프트웨어에 기초한 PostScript해석프로그람인 Ghostscript프로그람에 의해서 지원된다. 자기의 인쇄기가 지원되는가 보려면 http://www.cs.wisc.edu/~ghost/에 있는 Ghostscript홈페지를 조사해야 한다. 표 18-1에 Ghostscript가 지원하는 인쇄기들의 부분적인 목록을 주었다.

표 18-1. GhostScript가 지원하는 인쇄기

Canon BubbleJet BJ10e	HP DeskJet 682C
Canon BubbleJet BJ200	HP DeskJet 683C
Canon BubbleJet BJC-210(41)	HP DeskJet 693C
Canon BubbleJet BJC-240(3.33,43)	HP DeskJet 694C
Canon BubbleJet BJC-250(5.10)	HP DeskJet 850
Canon BubbleJet BJC-70(5.10)	HP DeskJet 855
Canon BubbleJet BJC-600	HP DeskJet 870Cse
Canon BubbleJet BJC-4000	HP DeskJet 870Cxi
Canon BubbleJet BJC-4100	HP DeskJet 890C
Canon BubbleJet BJC-4200	HP DeskJet 672C

Canon BubbleJet BJC-4300
Canon BubbleJet BJC-4550
Canon BJC-210
Canon MultiPASS C2500 color printer/fax/copier
Canon BJC-240
Canon BJC-70
Canon BubbleJet BJC-800
Canon BubbleJet BJC-7000
HP DeskJet
HP DeskJet Plus
HP DeskJet 500
HP DeskJet Portable
HP DeskJet 400
HP DeskJet 500C
HP DeskJet 540C
HP DeskJet 690C
HP DeskJet 693C
HP DeskJet 550C
HP DeskJet 560C
HP DeskJet 600
HP DeskJet 660C
HP DeskJet 680
HP DeskJet 1100C
HP DeskJet 500C
HP DeskJet 510
HP DeskJet 520
HP LaserJet 5
HP LaserJet 5L
HP LaserJet 6L
Oki OL410ex LED printer
NEC SuperScript 860
HP PaintJet XL300
HP DeskJet 1200C
HP DeskJet 1600C
Ricoh 4081 laser printer
Ricoh 6000 laser printer
Epson Stylus Color
Epson Stylus Color 11
Epson Stylus 500
Epson Stylus 600
Epson Stylus 800

주의 모든 인쇄기들은 Linux에서 작업하지 않는다. 몇개의 인쇄기들은 Microsoft Windows에서만 정보를 읽기 위해서 설정된다. 일부 USB인쇄기들은 아직까지 지원되지 못한다. 또한 http://www.linuxprinting.org/ 에 있는 Printer HOWTO에는 다루기 힘든 일부 인쇄기들로 작업하기에 대한 조언들이 포함된다. 이 Web싸이트는 또한 Linux에서 작업하는 인쇄기들에 대한 더 완벽한 목록도 포괄하고 있다.

printcap파일

printcap파일은 /etc 등록부에서 볼수 있으며 Linux 인쇄기구성의 핵이다. 이 파일은 Linux체계에 리용할수 있는 매 인쇄기에 대한 항목들을 포함한다. Linux 인쇄기데몬인 lpd는 적재될 때(일반적으로 기동시에) 관리되게 될 인쇄기에 대하여 알아 보기 위해 이 파일을 조사한다.

printcap파일의 기초항목은 다음과 같다.

```
# LOCAL
djet5001p|dj|deskjet:\
:sd=/var/spoo1/lpd/dj:\
:mx#0:\
:1p=/dev/1p0:\
:sh:
```

매 항목은 두점으로 구분되는 많은 마당을 가지고 있다. 항목에 한개이상의 행이 필요하면 거꿀빗선은 다음행에서 항목이 계속된다는것을 나타낸다.

이 실례에서는 3개의 가능한 이름 djet500lp, dj, deskjet로 된 인쇄기를 보여 주었

다. 이 인쇄기는 lpd가 인쇄를 기다리는 림시파일들을 보관할수 있는 완충등록부 /var/spool/lpd/dj를 가진다. 인쇄기는 첫 병렬포구(/dev/lp0)에 련결되고 머리부가 아닌 페지들을 인쇄하여야 한다. 이것은 모두 순수한 스크립트처럼 보여야 하며 그것은 일정한 범위에 대한것이다. printcap 파일은 수십개의 각이한 마당들을 가져야 한다. 이것들은 다음과 같은 printcap사용설명서페지에서 볼수 있다.

```
$ man printcap
```

만일 이 정보가 작업할 인쇄기를 얻는데 요구되였던것들의 전부라면 인쇄기들의 구성은 그리 나쁘지는 않다. 그러나 lpd는 본질적으로 보잘것 없다. 우에서 본 항목에서와 같이 lpd에 제공되는 아무자료나 직접 인쇄기에 전송된다. 만일 인쇄기가 제공된 자료를 인식하지 못한다면 알아 들을수 없는 말밖에는 아무것도 얻을수 없다. 다음과 같은 제기될수 있는 문제들을 고찰하자.

- PostScript인쇄기에 전송되는 ASCII본문파일은 인쇄되지 않는다.
- PCL인쇄기에 전송되는 ASCII본문파일은 그 파일에 있는 조종코드들이 조정되지 않는 한 정확한 형태로 인쇄되지 않는다.
- PCL인쇄기에 전송되는 PostScript파일은 지령으로 정의된 실지페지영상대신 PostScript지령들의 긴목록으로 인쇄된다.

이것들은 우에서 본것과 같이 요점적인 printcap항목을 리용하여 인쇄할 때 제기되는 문제들이다. 이것들은 인쇄려파프로그람들을 리용하여 해결할수 있다.

인쇄려파프로그람

인쇄려파프로그람들은 인쇄기에 전송되기전에 자료를 처리하는 특별한 프로그람들이거나 스크립트들이다. 례를 들어 PCL인쇄기가 있으면 인쇄기에 ASCII본문을 전송하기전에 처리하기 위한 하나의 스크립트를 작성함으로써 그것은 정확한 PCL 출력을 발생시키는 Ghostscript를 통하여 PostScript자료를 전송하는 정확한 형태와 다른 스크립트로 나타나게 된다. 이러한 조작을 수동으로 하려면 printcap파일에 다음과 같이 리용할수 있는 매 형태의 려파프로그람에 대한 다중인쇄기항목들을 가져야 한다.

```
# PCL Printer with ASCII filter
ascii-pcl :\
:sd=/var/spoo1/1pd/ascii-pcl:\
:mx#0:
:1p=/dev/1p0:\
:sh:\
:if=/var/spoo1/lpd/ascii-pcl/fi1ter

# PCL Printer with PostScript filter
ps-pc1:\
:sd=/var/spoo1/1pd/ps-pc1:\
:mx#0:
:1p=/dev/1p0:\
:sh:\
:if=/var/spoo1/lpd/ps-pcl/fi1ter
# PCL Printer with no filter
```

```
pc1:\
:sd=/var/spool/1pd/pc1:\
:mx#0:
:1p=/dev/1p0:\
:sh:\
```

이 려파프로그람들은 정확한 출력을 발생시키지만 인쇄기처리에 복잡성을 야기시킨다. 사용자들은 자기들의 응용프로그람에 의해 발생되는 출력형태(명백한 본문, PostScript, PCL)가 어떤것인지 알아야 하며 그에 맞게 정확한 인쇄기를 선택해야 한다. 실제로 한개의 인쇄기가 있을 때 려파프로그람들은 리용할수 있는 3개의 물리인쇄기들이 있다고 암시하기때문에 혼돈될수 있다.

이 혼동은 위력한 려파처리프로그람들로 처리된다. 인터네트로부터 내리적재할수 있는 위력한 려파프로그람들은 가장 많이 지원되는 인쇄기들에 대한 모든 printcap구성을 조정하며 인쇄기에 전송되는 자료들의 형태를 결정할수 있는 려파프로그람들을 제공하며 그 기초우에서 정확한 려파를 진행한다.

APS 인쇄려파프로그람체계

APS인쇄려파프로그람체계 위력한 인쇄려파프로그람묶음을 소개한다. http://www.apsfilter.org로부터 APS인쇄려파프로그람의 최신판본을 내리적재할수 있다. 이것을 쓸 당시의 APS인쇄려파프로그람의 공개된 판본은 6.1.1이다. 이 장에서 내리적재해야 할 파일은 apsfilter-6.1.1.tar.gz이다. 여기서 서술된 조작들은 다른 판본들에서와 다르다. 의문되는 점이 있으면 Linux체계우에 이 파일을 내리적재하고 푼 다음에 README파일을 참고해야 한다.

이 쏘프트웨어를 설치하려면 /usr/local과 같은 론리적인 장소에서 문서고를 풀고 꺼내기 위한 지령 tar를 리용해야 한다. 그러자면 풀어 진 파일들을 보관하는 장소로서 등록부를 하나 만들어야 한다. 다음의 지령들은 /tmp에 초기의 문서고를 내리적재하였다고 가정한다.

주의 뿌리사용자들은 이 단계를 거쳐야 한다.

```
# cd /usr/local
# tar xzvf /tmp/apsfilter6.1.1.tar.gz
# cd apsfilter
```

인쇄기가 작업하는 APS를 구성하는 다음단계는 다음과 같은 프로그람묶음이 들어있는 SETUP스크립트를 실행하는것이다.

```
# ./SETUP
```

APS인쇄려파프로그람은 Ghostscript 프로그람묶음으로 작업한다. 만일 Ghostscript의 충분히 개선된 판본이 없으면 SETUP프로그람은 사용자들에게 이것을 알게 한다. 주의에도 불구하고 APS인쇄려파프로그람 6.1.1은 Red Hat Linux 7.1에 있는 Ghostscript(5.5)판본으로 작업하지 않는다.

주의 만일 최신인쇄기구동프로그람들이 요구되면 Ghostscript최신판본을 내리적

재해야 하는데 그것은 http://www.cs.wisc.edu/~ghost/에서 참고해야 한다. Ghostscript를 이미 써보았다면 그 이름이 Aladin Ghostscript로부터 APFL Ghostscript로 변경되였다는것을 명심하여야 한다.

이것은 그림 18-1에서 보여 준 환영화면이 나타나게 한다. y를 입력하여 APS인쇄려파프로그람에 대한 GNU일반공개사용허가의 조항들을 입수한 다음 Enter를 누른다. 우편주소들에 대한 재촉문에 입력한 다음 SETUP프로그람을 그림 18-2에 보여 준 Installation Program화면에로 안내한다.

```
Hi, welcome to the apsfilter setup and thanks for using apsfilter!

Before we begin with the setup, I'd like to make you familiar
with the apsfilters license:
1.  apsfilter follows the GNU public license (GPL), see COPYRIGHT
2.  and this little "Postcard License":

        " I'd like to get a postcard from you! I'm interested in,
        who is using apsfilter, where you live, and where in the
        world apsfilter is doing it's job. "
Please send me an e-mail to request-snailmail@klemm.gtn.com to
get my postal address; we 'll get there later in this setup ...

Accept license [Y|y|J|j|N|n]? _
```

그림 18-1. APS 인쇄려파프로그람의 기동

```
=====================================================
            A S P F I L T E R   V 6 . 1 . 1
            *** The Unix Print Solution ***
=====================================================

        copyright Andreas Klemm, 1993-2001

            andreas@apsfilter.org

          http://www.apsfilter.org/

     ----------------------------------------
        for Unix systems with BSD compatible
           line printer scheduler (lpd)
                 or with LPRng
       FreeBSD / NetBSD / OpenBSD / BSDI
         Linux and other flavours of Unix
     ----------------------------------------

        ++++ INSTALLATION PROGRAM ++++
          [ press <RETURN> to continue ]
```

그림 18-2. APS 인쇄려파프로그람의 설치

계속하려면 Enter를 눌러야 한다. Apsfilter Setup 스크립트의 기능들을 설명하는 화면을 읽고 계속하기 위하여 다시 Enter건을 누른다. 다음단계는 그림 18-3에서 보여 준 것처럼 인쇄기구성파일을 보관하는 구성등록부를 설정하는것이다. /etc/apsfilter 기정선택항목은 대체로 다른 Red Hat Linux 7.1구성파일들로 구성되는데 그 항목들에 대한 선택권은 사용자에게 있다.

```
========================================
      Setting up configuration directory
========================================

Please enter the path of the directory you want to keep the
printer configuration files in. This directory will be created
if it doesn't already exist.

Common location are:

      /etc/apsfilter                      /usr/etc/apsfilter
      /usr/local/etc/apsfilter            /opt/etc/apsfilter
      /usr/plus/apsfilter/etc             /opt/local/apsfilter/etc

and so on.

You might run into trouble if you enter the apsfilter
installation directory here (but an "etc" subdirectory is okay).

Default: /etc/apsfilter

You choice:
```

그림 18-3. 구성등록부의 설정

Enter를 누르면 재촉문이 나타난다. 스크립트는 적용할수 있는 등록부의 허가들을 검색하고 /etc/printcap.old에 현재 인쇄기구성파일을 여벌복사함으로써 계속한다.

일단 선택을 만들었다면 그림 18-4에 보여 준 Apsfilter Setup Main Menu에로 안내된다. 인쇄기를 구성하려면 1, 2항목들을 처리해야 한다. 또한 항목 3～5의 항목들로 검사페지를 설정하면 T항목을 실행할수도 있다.

```
=============================================================
  APSFILTER SETUP                          -- MAIN MENUE --
=============================================================

                                            currently selected
-------------------------------------------------------------
(D)  Available Device Drivers in your gs binary
(R)  Read Ghostscript driver documentaion         (driver.txt)
(1)  Printer Driver Selection                     [ ]
(2)  Interface Setup                              [ ]

For printing the test page:
(3)  Paper Format (mandatory)              [ ]
(4)  Print Resolution in "dots per inch"   [ ]
(5)  Toggle Monochrome/Color (1bpp=b&w)    [ ]
(T)  Print Test Page, on local or Windows remote prt.(after 1-5)
(V)  View perf.log (times off print attempts)

(A)  Abort installation (don't do anything)
(I)  ==> Install printer with values shown above - repeat this
         step for installing multiple printers
(Q)  ==> Finish installation

Your choice?
```

그림 18-4. Apsfilter Setup Main Menu

첫 두개의 차림표항목들인 D, R는 Ghostscript설치로 할당된 가능한 장치구동프로그람들과 Ghostscript구동프로그람문서에 대한 페지를 목록화한다. 선택항목 1과 2는 다음과 같이 인쇄기를 구성한다.

인쇄기구동프로그람선택

Main Menu로부터 선택항목1을 선택한다. APS Print Filter 편의프로그람은 그림 18-5에서와 같이 8개의 각이한 형태의 구동프로그람들을 포함한다. 현대적인 인쇄기들은 현실성 있는 PostScript인쇄기이든가 일부형태의 Ghostscript구동프로그람을 요구할수

있다. 자기의 인쇄기를 잘 모르겠다면 이 선택항목들중 몇가지를 열람해 보아야 한다. 인쇄기가 현실성 있는 PostScript가 아니라면(인쇄기에 대한 문서를 참고해야 한다.) 기타 선택항목가운데서 한 항목으로 목록화된 인쇄기를 볼수 있다.

주의 세가지 부류 즉 gimp-print, pcl3, IBM Omni는 자기의 목록화된 인쇄기에 특수한 Ghostscript지원을 제공한다. 만일 그 부류들을 모두 열람한후에도 인쇄기에 알맞는 부류를 아직 찾지 못했으면 http://www.linuxdoc.org/HOWTO/Printing-HOWTO/index.html에서 Printing HOWTO를 참고해야 한다.

그림 18-5. APS인쇄기구동프로그람선택차림표

- 항목 1은 현실성 있는 각이한 해상도의 PostScripts인쇄기들을 선택하게 한다. 인쇄기의 해상도는 인쇄기가 평방인치당 혹은 dpi당 조종할수 있는 점들의 수로 결정된다. Red Hat Linux 7.1에는 300～2880범위의 dpi를 조종할수 있는 PostScript구동프로그람이 있다.
- 항목 2는 Ghostscript에 의해서 그대로 제공되는 150개이상의 인쇄기에서 선택할수 있게 한다.
- 항목 3은 gimp형식의 인쇄접속방식에 기초하는데 그것은 각종 Canon, Epson, Lexmark, HP인쇄기에 높은 질을 가진 도형을 지원한다.
- 항목 4는 기본적으로 HP DeskJet 및 DeskJet색인쇄기를 지원한다.
- 항목 5는 항목 4를 대신한다. 그것이 《경험에 기초》하므로 이 구동프로그람으로부터 얻은 출력은 더 좋을수도 있고 전혀 동작하지 않을수도 있다.
- 항목 6은 IBM Omni인쇄기구동프로그람으로 제공된 250대이상의 인쇄기들을 위한것이다.
- 항목 7은 HP DeskJet 800과 900계렬형의 색인쇄기들을 위한것이다.
- 항목 8은 HP DeskJet 700, 800, 1000계렬형의 색인쇄기들을 위한것이다.

자기의 인쇄기에 가장 밀접히 일치하는 항목을 선택하면 긴 목록을 빠짐없이 조종하는 항목들을 가진 차림표를 보게 된다. Enter를 누르면 인쇄기이름을 찾을 때까지 목록

을 훑어 본다. 인쇄기와 련관이 있는 번호를 기록하고 재촉문이 나타나면 그것을 입력해야 한다. 대리방식으로 Printer Driver Selection차림표로 돌아 가기 위해서 0을 선택하고 Enter 를 누를수 있다.

일단 선택항목이 확증되면 스크립트는 Apsfilter Setup Main Menu를 복귀하게 한다. 그림 18-6에서 보여 준것과 같이 새 인쇄기장치는 [ljet4]인데 그것은 HP LaserJet 4L인쇄기에 해당한다.

```
=========================================================
 A P S F I L T E R   S E T U P              --MAIN MENUE--
=========================================================

                                              currently selected
---------------------------------------------------------
(D)   Available Device Driver in your gs binary
(R)   Read Ghostscript driver documentation          (devices.txt)
(1)   Printer Driver Selction                        [ ljet4 ]
(2)   Interface Setup                                [ ]

For printing the test page:
(3)   Paper Format (mandatory)                       [ ]
(4)   Print Resolution in "dots per inch "           [ default ]
(5)   Toggle Monochrome/Color (1bpp=b&w)             [ default ]
(T)   Print Test Page, on local or Windows remote prt.(after 1-5 )
(V)   View perf.log ( times of print attempts )

(A)   Abort installation ( don 't do anything)
( I)   ==> Install printer with values shown above - repeat this
           step for installing multiple printers
(Q)   ==> Finish installation

Your choice?
```

그림 18-6. Apsfilter구성차림표

주의	많은 인쇄기들은 리용가능한 한개이상의 구동프로그람들을 가진다. 례를 들어 HP LaserJet 4L인쇄기는 Ghostscript구동프로그람과 gimp형식의 인쇄구동프로그람을 다 같이 가진다. 시험과정에 나타나는 통보문이나 오유들은 자기의 요구에 잘 맞는 구동프로그람들을 찾을것을 요구할수 있다.

알려두기	이 차림표에서 자기의 인쇄기와 관련된 해상도를 찾아야 한다. 만일 인쇄기문서에 제시된것과 차이난다면 Apsfilter기본차림표의 항목 4 즉 Print Resolution에서 설명한것처럼 그것을 재설정할수 있다.

대면부설정

인쇄기와 관련된 대면부를 정의하거나 변경시킬수 있다. 다음단계는 인쇄기의 위치를 구성하는것이다. 그것은 국부적인 병렬포구나 직렬포구에 접속될수 있다. 그것은 각이한 형태의 망에서 원격조종인쇄기에도 접속될수 있다. 이 구성을 구체적으로 설정하려면 그림 18-7에서 보여 준 Apsfilter Interface Setup화면을 소개하는 Apsfilter Setup Main Menu에서 항목 2를 선택해야 한다.

Interface Setup차림표에서 볼수 있는것처럼 5가지 각이한 부류의 인쇄기결합들을 구성할수 있다.

Local Parallel/USB 병렬포구는 일반적으로 25핀접속구로 자기의 콤퓨터와 결합된다. Linux쏘프트웨어는 앞에서 설명한것처럼 /dev/lp0, /dev/lp1와 같은 장치들에 결합된다.

그림 18-7. Apsfilter Interface Setup화면

주의	일부 USB인쇄기들은 Linux병렬포구형장치를 리용할수 있게 설정될수 있다. 병렬포구들은 /dev/lp0 및 /dev/lp1과 같은 장치들과 결합되면 USB인쇄기포구들은 /dev/usb/lp0및 /dev/usb/lp1과 같은 장치들과 결합된다.

Local Serial 직렬포구는 일반적으로 9핀접속구로 자기의 콤퓨터와 결합된다. Linux쏘프트웨어는 이 장의 후에 설명되는 /dev/ttyS0및 /dev/ttyS1과 같은 장치들에 결합된다.

Unix/Network Printer 이것은 국부망이나 기타망에 있는 각이한 Linux콤퓨터 혹은 Unix콤퓨터에 련결된 인쇄기의 접속구를 설정하게 한다.

Windows/NT(Samba) 이것은 국부망이나 기타망우에 있는 Microsoft Windows 95/98/Me/NT/2000콤퓨터에 련결된 인쇄기에 대한 접속구를 설정하게 한다.

AppleTalk 이것은 국부망이나 기타망에 있는 Apple콤퓨터에 련결된 인쇄기에 대한 접속구를 설정하게 한다.

Apsfilter Setup Main Menu로 돌아 가려면 하나를 선택하고 Enter건을 눌러야 한다.

검사페지만들기

Apsfilter Setup 를 끝내기전에 검사페지로 자기의 작업을 검색하는것을 방조한다. 그러나 검사페지를 인쇄하기전에 종이의 크기나 형식, 인쇄기의 해상도를 구성하고 인쇄기가 흑색인쇄기인가 아니면 색인쇄기인가를 결정해야 한다.

종이크기를 설정하려면 Main Menu의 항목 3 Paper Format를 선택해야 한다. 이것은 다음과 같은 5가지 선택들을 제공한다.

- DIN A4는 8.27×11.69(210×297㎜)인 측정단위를 가진 표준종이크기이다.
- DIN A3은 11.69×16.54(297×420㎜)인 측정단위를 가진 표준종이크기이다
- US Letter는 8.5×11(215.9×279.4㎜)인 측정단위를 가진 표준US Letter종이크기이다

- US Legal은 8.5×14(215.9×355.6㎜)인 측정단위를 가진 표준US Letter종이 크기이다
- US Ledger는 11×17(279.4×431.8㎜)인 측정단위를 가진 더 큰 크기이다.

인쇄기에 가장 적합한 종이를 선택하고 Enter를 누른다.

Apsfilter Setup Main Menu를 다시 보고 항목 4 Print Resolution을 선택한다. 만일 인쇄기구동프로그람을 선택했을 때 기정해상도가 만족되면 더 변경하지 않아도 된다. 한편 이 차림표는 많은 전형적인 설정들을 선택하거나 자체의 인쇄해상도설정을 전용화하게 한다.

해당한것을 선택하고 Enter를 누른다. 새 인쇄기를 구성하고 있다면 기정색의 질을 변화시킬수 있다. 그러자면 항목 5 Toggle Monochrome/Color를 선택해야 한다. 이 차림표의 현재판본은 색을 1bpp(흑색)∼ 32bpp(색)범위에서 선택하게 한다. 선택한 다음 Enter를 누른다.

구성의 검사

일단 Apsfilter구성을 완성하고 필요한 케블들을 련결했다면 작업을 검사할수 있다. Apsfilter Setup Main Menu로부터 **T**를 누르고 재촉문에 응답해야 한다. 특히 고해상도 인쇄기 혹은 색인쇄기가 있을 때 검사페지설정이 때때로 제기된다. 성공하면 Screen Testing으로 페지를 보거나 각이한 크기의 서체표본들을 볼수 있다.

일단 구성이 만족하면 Apsfilter Setup Main Menu에서 작성한 설정들을 설치하기 위하여 **I**를 눌러야 한다. 기정인쇄기와 《렬》인쇄기들에 대한 재촉문을 조사한후에 /etc/printcap파일에서 설정들을 작성한다. 그다음 Apsfilter Setup Main Menu에서 **Q**를 눌러서 끝낼수 있다.

구성후에

구성후에 다음과 같이 보이는 printcap파일을 가질수 있다.

```
# /etc/printcap
#
# DO NOT EDIT! MANUAL CHANGES WILL BE LOST!
# This file is autogenerated by printconf-backend during lpd init.
#
# Hand edited changes can be put in /etc/printcap.1ocal, and will be
# included.

##########################################################################
## Everything below here is included verbatim from /etc/printcap.1ocal##
##########################################################################
# printcap.local
# This file is included by printconf's generated printcap,
# and can be used to specify custom hand edited printers.
# APSI_BEGIN:printerl
# - don't delete start label for apsfilter printer1
```

```
# - no other printer defines between BEGIN and END LABEL

1p|Printer1 auto:\
:1p=/dev/1p0:\
:if=/etc/apsfi1ter/basedir/bin/apsfi1ter:\
:sd=/var/spool/lpd/lp:\
:lf=/var/spool/lpd/lp/1og:\
:af=/var/spool/lpd/lp/acct:\
:mx#0:\
:sh:
rawl |Printer1 raw:\
:lp=/dev/lp0:\
:if=/etc/apsfi1ter/basedir/bin/apsfi1ter:\
:sd=/var/spool/lpd/rawl:\
:lf=/var/spool/lpd/rawl/1og:\
:af=/var/spool/lpd/rawl/acct:\
:mx#0:\
:sf:\
:sh:
# APS1_END - don't delete this
```

경고 Apsfilter를 리용하면 제15장에서 본 Printconf편의프로그람을 리용하지 않아도 된다. 만약 그렇지 않으면 Apsfilter로 작성한 설정들은 잃어 버리게 된다.

Linux로 리용하기 위하여 구성하는 인쇄기에 무관계하게 Apsfilter체계는 다음의 인쇄대기렬을 작성하게 한다.

lp | Printer1 auto 파일형태를 알아 내고 그것을 적당히 처리하게 되는 자동적인 려파프로그람

raw 리용하고 있는 쏘프트웨어가 인쇄기에 필요한 본래형태로 출력을 발생시킬 때 리용할수 있는 려파되지 않는 인쇄대기렬

주의 이미 본것처럼 기초적인 Apsfilter구성파일은 설치과정에 선택했던 등록부에 보관된다. 기정**Apsfilter**등록부는 **/etc/apsfilter**이다. Apsfilter 구성파일을 다시 실행하고 기정 Apsfilter 등록부를 리용했다면 **/etc/apsfilter/basedir /SETUP**지령을 리용해야 한다.

인쇄

이제는 인쇄할 준비가 되였다. lpr지령으로 이것을 진행할수 있다. –p기발은 리용할 인쇄기렬을 정의한다. 례를 들어 지령

$ lpr –Praw /etc/printcap

은 raw렬이나 정확한 본문인쇄렬로 printcap파일을 찍어 낸다. 한편 지령

```
$ lpr –Plp /etc/printcap
```

은 자동적인 려파프로그람으로 같은 파일을 인쇄한다.

주의	우에서 본 lpr지령에는 –P와 인쇄기의 이름사이에 공백이 없다.

모뎀

모뎀은 Linux의 작업환경을 얻기 위한 가장 간단한 주변장치에 속한다. 많은 경우 Linux는 자동적으로 모뎀을 구성하는 끼운즉시동작특성들을 리용한다. 일반적으로 외장모뎀은 리용가능한 직렬포구에 끼워 질수 있다. 한편 표준내장모뎀은 Linux체계에서 설치하기 쉽고 실행하기 쉬워야 한다.

주의	내부모뎀이 아닌 모든 모뎀들은 Linux에서 지원된다. 한개 모뎀이 있으면 그것을 시험해야 한다. 새 모뎀을 사려고 할 때 여분의 비용은 외부모뎀을 살만한 가치가 있어야 하는데 그것은 보관할수 있는 구성의 시끄러운 절차들에 따른다. 만일 내장모뎀을 선택하면 winmodems라고 하는 쏘프트웨어모뎀을 피해야 하는데 이것은 흐름조종을 완성하기 위한 특수화된 Windows쏘프트웨어를 리용한다. 그러나 이러한 Windows계모뎀들을 리용해도 전부가 없어지는것은 아니다. 즉 http://www.linmodems.org에 Windows계모뎀들을 위한 구동프로그람들도 있다.

이 절에서는 Red Hat Linux 7.1하드웨어검출도구 Kudzu를 고찰한다. 그러나 이 도구가 모뎀을 검출하지 못하면 모뎀을 적당히 구성하기 위하여 Linux가 포구들을 조정하는 방법을 알아 둘 필요가 있다. 마지막으로 이 절에서는 모뎀을 리용하기 위한 몇가지 간단한 방법들을 고찰한다.

모뎀검출

일부 Linux배포물은 하드웨어검출도구를 가지고 있다. Red Hat Linux 7.1에 포함된 도구들중의 하나가 Kudzu인데 이것은 후에 하루에 한피트이상 자라는 동아시아포도로 명명되였다. 다음의 지령으로 Kudzu를 실행하는 뿌리사용자방식에로 넘어 갈수 있다.

```
# /usr/sbin/kudzu
```

만일 모든것이 설치된 하드웨어관련목록들에 따라 구성되면 Linux는 한동안 지연이 생긴다. 그다음 지령행으로 복귀된다. Kudzu는 문서화된 구성에 기초한 하드웨어에 대한 신호들을 전송한다. 그러나 최근에 하드웨어를 변경했다면 그림 18-8에서 보여 준것과 같은 화면이 나타나게 된다.

Kudzu가 검출한것을 보려면 임의의 건을 눌러야 한다. 그 실례를 그림 18-9게서 보여 준다.

Kudzu가 검출되였기때문에 이 경우에 모뎀은 전용포구우에 있으며 모뎀을 체계에

추가하려면 필요한것만을 선택해야 한다.

Kudzu와 같은 도구가 새로운 하드웨어를 조사할 때 이미 검출된 하드웨어의 자료기지를 고려하면서 하드웨어를 검색하는데 그 자료기지는 /etc/sysconfig/hwconf파일에 보관되여 있다.

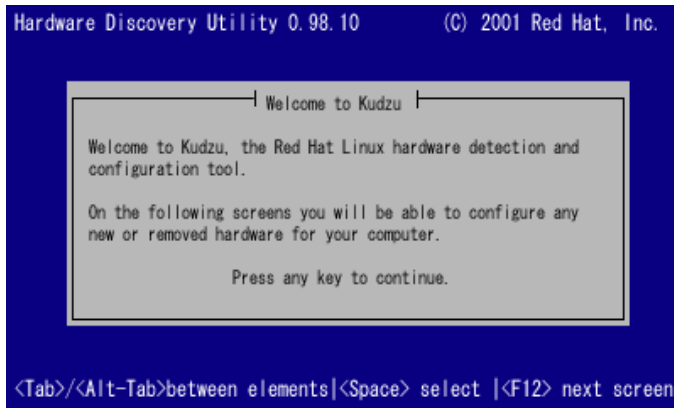



그림 18-8. Kudzu소개화면

이 파일이 아직 작성되지 않았다면 Kudzu는 /etc/moduls.conf, /etc/sysconfig/network-scripts, /etc/X11/XF86 Config 혹은 /etc/X11/XF86Config-4구성파일에서 검출된 하드웨어를 검색한다.

그림 18-9. Kudzu는 무엇인가 새로운것을 검출하였다.

많은 경우에 제22장 《Linux를 인터네트에 접속》에서 본 인터네트결합도구에로 직접 넘어 갈수 있다. 그러나 Kudzu나 Linux배포에 있는 해당한 편의프로그람을 만들수도 있다. 이때 새로운 모뎀이나 다른 하드웨어를 검출할수 없거나 잘못된 포구에 설정할수 있다. 그 경우에 Linux포구로 작업하자면 다음절에서 보게 되는 도구들이 있어야 한다.

일러두기 kudzu는 Red Hat Linux 7.1을 기동 혹은 재기동할 때 자동적으로 실행된다.

그것이 하드웨어구성에 대한 변경들을 검출하면 그림 18-8과 같은 화면을 보게 된다. 새 하드웨어를 구성하는 Kudzu에 들어 가려면 30초내에 임의의 건을 눌러야 한다. 이 시간내에 임의의 건을 누르지 못하면 Linux의 기동과정은 계속된다. 항상 Linux가 기동을 완료한후 Kudzu를 기동할수 있다.

Linux포구의 리해

Linux환경에서 모든 물리적인 주변장치나 련결포구는 정의된 등록부 /dev 에 있는 한개이상의 파일들에 관계된다. 이것은 하드디스크구동기들, CD-ROM구동기들, 병렬포구들, 직렬포구들을 포함한다.

이에 대한 개념은 리해하기 쉽다. 한가지 실례로 하드구동기들을 보기로 하자.

매 IDE하드구동기나 CD-ROM은 hdx로 명명되는데 여기서 x는 기본IDE모선의 첫번째 구동기에 대해서는 a, 기본IDE모선의 두번째 구동기에 대해서는 b, 두번째 IDE모선의 첫번째 구동기에 대해서는 c, 두번째 IDE모선의 두번째 구동기에 대해서는 d와 같은 문자이다.

주의 SCSI하드나 CD-ROM구동기들은 sdx와 비슷하게 명명된다.

이외에 구동기우의 매 구획은 /dve항목에 소속된다. 례를 들어 초기의 종속구동기에 있는 두번째 구획은 /dev/hdb_2이며 두번째 주구동기의 첫번째 구획은 dev/hdc_1이다.

병렬포구와 Linux

지금까지 간단한 개념을 보았다. 이제부터 병렬포구를 고찰하자. DOS와 Windows 세계에서 병렬포구는 LP_1:, LP_2:, LP_3: 등으로 치부되였다. 대부분의 PC들에는 일반적으로 한개의 병렬포구만이 있으므로 유일하게 LPT_1:를 고찰해야 한다.

Linux세계에서 병렬포구들은 장치파일인 lpx에 대응하는데 여기서 x는 포구의 번호이다. 여기에 중대한 차이점이 있다. 즉 Linux는 포구를 1이 아니라 0부터 세므로 일반적으로 LPT_1:는 /dev/lp0이고 LPT_2:는 /dev/lp1이여야 한다. DOS에서 리용할수 있는 한개의 포구만이 필요한 경우도 있을수 있는데 이것은 LPT_1:이 아니다. 이러한 경우 이 포구는 /dev/lp0으로 작성되는데 dev/lp0는 항상 첫번째로 리용할수 있는 병렬포구이고 /dev/lpl은 두번째로 리용할수 있는 포구이다.

직렬포구와 Linux

Linux에서 직렬포구들은 일반적으로 개개가 두가지 장치파일들과 결합된다. 즉 하나는 수출접속구에 대한것이고 다른 하나는 수입접속구에 대한것이다. 수출접속구에 대하여 보자. 즉 ttySx가 있는데 여기서 x는 0부터 시작하는 번호이다. 그러므로 Linux에서 DOS의 COM_1:은 /dev/$ttyS_0$으로 작성하고 COM_2:은 /dev/$ttyS_1$로 작성한다. Linux의 일부 낡은 판본들에서 이 포구들은 dev/cua0과 /dev/cua1로 작성할수 있다. 이 할당들을 표 18-2에서 보여 주었다.

표 18-2. DOS와 Linux직렬포구

DOS이름	Linux장치파일들	낡은 판본의 장치파일들
COM_1:	/dev/$ttyS_0$	/dev/cua_0
COM_2:	/dev/$ttyS_1$	/dev/cua_1
COM_3:	/dev/$ttyS_2$	/dev/cua_2
COM_4:	/dev/$ttyS_3$	/dev/cua_3

그러나 이 장치구동기들은 대부분 사람들에게 있어서 기억하기 힘들다. 다행히도 사용자가 모뎀을 구성할 때 대부분의 Linux배포들은 즉시 /dev/modem파일과 현재장치사이 련결을 확립된다. ls –1 /dev/modem지령의 출력으로 이것을 검색할수 있다.

```
lrwxrwxrwx 1   root   10 Feb      3  19:21   /dev/modem -> /dev/ttys0
```

주의 /dev/cua_1, /dev/cua_2을 위한 지원과 그우에서는 낡은것들과의 량립성만을 제공하며 결국 이 파일을 지원하지 않는다.

USB포구와 Linux

리론적으로는 32USB모뎀들을 Linux 콤퓨터에 접속할수 있다. USB모뎀들을 리용하기 위하여 알맞는 USB모듈들이 핵심부에 있는가를 확인해야 한다. 이에 대해서는 제20장 《Linux핵심부의 재콤파일》에서 고찰하게 된다. 일부 USB모듈들은 Red Hat Linux 7.1을 비롯하여 최신판배포들에 이미 포함되여 있다.

USB모뎀에는 또한 Communication Device Class Abstract Control Modules 또는 acm.o가 필요하다. 그것이 이미 핵심부의 부분이 아니면 그것을 현재구성에 포함시킬수 있다. Red Hat Linux 7.1에서는 이 모듈을 핵심부로 콤파일해야 한다. 수정한 핵심부에 대한 더 상세한 정보는 제20장에서 참고할수 있다. 일단 이것이 완성되면 다음의 지령을 실행할수 있다.

```
# /sbin/insmod /lib/modules/2.4.2-2/핵심부/drivers/usb/acm.o
```

insmod나 핵심부판본들이 다르면 적당히 바꾼다. Linux배포물에 실지로 있는 등록부는 바꿀수 있다.

알맞는 장치들은 이미 /dev/usb등록부에 있어야 한다. ls –1 /dev/usb/ttyACM*지령으로 이것을 검색한다. 출력은 다음과 같이 나타나야 한다.

```
crw-rw----      1 root root    188,   0 Aug 24 09:00 /dev/usb/ttyACM0
crw-rw----      1 root root    188,   0 Aug 24 09:00 /dev/usb/ttyACMl
crw-rw----      1 root root    188,   0 Aug 24 09:00 /dev/usb/ttyACM2
crw-rw----      1 root root    188,   0 Aug 24 09:00 /dev/usb/ttyACM3
```

이 형태의 출력이 필요 없으면 다음과 같은 지령을 실행시킨다.

```
mknod /dev/usb/ttyACM0 c 166 0
```

필요한 USB모뎀을 얻기 위하여 $ttyACM_1$, $ttyACM_2$ 등을 교체하면서 이 지령을 반복실행한다.

모뎀준비

모뎀이 정확히 작업하도록 하기 위해서는 몇가지 사항을 만족시켜야 한다.

- PC에 정확히 접속되여야 한다.
- 널리 알려 진 직렬포구를 리용하여야 한다.
- 다른 장치와 모순되는 IRQ 또는 I/O주소를 가지지 말아야 한다.

모뎀을 PC에 접속

모뎀의 접속은 아주 간단하다. 외장모뎀이 있으면 이미 있는 직렬포구중의 한포구와 마찬가지로 강력한 공급포구에 그것을 접속해야 한다. 빈 포구들이 없으면 심사숙고해서 직렬카드를 구입해야 한다. 즉 PC나 모뎀판매자와 이것을 상론해야 한다. 그러나 일반적으로 PC사용자들은 두개의 외장직렬포구들만을 요구한다. 즉 하나는 마우스를 위한것이고 다른 하나는 모뎀을 위한것이다. 보통 실천에서는 마우스를 첫번째 직렬포구(COM_1:, $/dev/ttyS_0$, $/dev/cua_0$)에 접속하고 모뎀을 두번째 직렬포구(COM_2:, $/dev/ttyS_1$, $/dev/cua_1$)에 접속한다. Red Hat Linux 7.1에서는 대체로 이것을 /dev/modem파일에 련결시킨다.

한개의 USB모뎀을 리용하고 있으면 그것은 장치를 /dev/modem파일에 련결시켜야 한다. 실례로 $/dev/usb/ttyACM_0$을 리용하고 있으면 ln –s /dev/modem $/dev/usb/ttyACM_0$지령을 실행시켜야 한다.

보통 내장모뎀은 PC의 빈 홈에 배치되여야 한다. 빈틈없이 PC를 개방하려면 모뎀을 설치하기 위해서 PC와 모뎀을 구입했던 상점으로부터 전문가를 받는것을 생각해 보아야 한다.

직렬포구선택

외장모뎀을 설치할 때 간단히 모뎀을 어느 물리포구에 접속하겠는가를 결정하여 직렬포구를 선택할수 있다. 여기서 같은 직렬포구를 리용하는 내장장치를 설치하지 않았다는것을 확인하는것이 중요하다. PC에 어떤 하드웨어를 설치하지 않았다면 외장접속구들의 포구와 같은 포구를 리용하는 어떤 초기 설치된 내장장치들이 있는가를 보기 위하여 PC의 문서를 참고해야 한다. 그러나 일반적으로 이미 설치된 내장모뎀이 없으면 리용할수 있는 외장직렬포구들의 한 포구와 상반되는 내장장치를 가질수 있다고 기대할수 없다. 내장모뎀의 설치는 좀 많은 작업을 요구한다.

주의 많은 낡은 외장모뎀들은 기능에 맞는 전용직렬포구를 리용하기 위해서 구성되여야 한다. 일부 경우에 특수한 직렬포구를 정의하기 위해서 모뎀카드에 있는 DIP스위치들이나 이행선들을 물리적으로 조정하여야 한다. 이것을 진행하는 방법에 대해서는 지령들에서 요구된것처럼 모뎀의 문서를 참고해야 한다.

외장모뎀에서와 같이 이미 리용한 외부장치와 일치시키는 포구를 선택하지 못한다는 것을 확인해야 한다. 례를 들어 이 포구에 대하여 외장접속기에 접속된 마우스가 있으면 확정적으로 모뎀이 첫번째 직렬포구를 리용하지 못하게 하여야 한다. 세번째나 네번째

직렬포구들은 보통 내장모뎀에 대한 가장 좋은 선택이다. 첫번째 직렬포구는 /dev/ttyS0에, 두번째 직렬포구는 /dev/ttyS1에 대응한다는것을 명심해야 한다. 다음절에서는 직렬포구의 구체적인 차례와 선택에 대하여 보게 된다.

IRQ 및 I/O 주소

매 직렬포구와 련관된것은 한개의 새치기요구(IRQ)주소와 한개의 입력/출력(I/O)주소이다. 수로 정의된 이 정보는 PC와 조작체계가 물리장치가 항상 자료나 정보를 제공하거나 체계에 대한 주의를 요구하고 있다는것을 안다는것을 확증한다.

표 18-3에는 4개의 공동직렬포구와 그것과 련관된 IRQ 및 I/O주소들을 주었다.

표 18-3. 직렬포구 IRQ와 I/O주소

DOS이름	Linux장치파일	IRQ	I/O주소
COM1:	/dev/ttyS0	4	0×3f8
COM2:	/dev/ttyS1	3	0×2f8
COM3:	/dev/ttyS2	4	0×3e8
COM4:	/dev/ttyS3	3	0×2e8

첫번째와 세번째 포구들은 같은 IRQ를 공유하고 두번째와 네번째 포구도 역시 같은 IRQ를 공유한다는데 주목을 돌려야 한다. 이것은 고찰되여야 하는 문제점을 제시한다. 포구1과 포구3이 같은 IRQ를 공유하지만 본질적으로는 다른 I/O주소를 가진다는 사실은 그것이 동일하게 리용될수 없다는것을 의미한다. 즉 이 두개의 포구들에 대하여 동시에 리용할수 있는 장치들을 가질수 없다.

례를 들어 이것은 모뎀이 리용되고 있는 동안 마우스를 절대로 리용하지 않는 한 첫번째 포구에 마우스를 가질수 없으며 세번째 포구에 모뎀을 놓을수 없다는것을 의미한다. X Windows를 리용하면 이것은 아주 믿어 지지 않는 스크립트이다. 그러므로 내장모뎀을 설치할 때 IRQ새치기를 만들지 못한다는데 주의를 돌려야 한다. 만일 마우스를 첫번째 직렬포구에 리용하면 마우스와 함께 모든 가능한 새치기들을 소거하기 위해서 대체로 네번째 포구에 내장모뎀을 놓으려고 한다.

마지막으로 한개이상의 고찰이 있다. 내장모뎀은 흔히 카드에 4개 직렬포구의 표준들과 구별되는 특수한 IRQ와 I/O주소들을 구성하게 한다. 이것은 다른 방법으로는 모뎀제작품을 간단히 만들수 없는 체계에 이미 아주 많은 직렬장치들을 가지고 있을 때에만 수행될수 있다.

PnP(Plug-and-Play)모뎀

PnP주기판과 Windows 95로 기동하는 Microsoft조작체계의 도입은 다른 조작체계와 관련된 내장기판을 구성하는 몇가지 문제를 제기하였다. 많은 PnP장치들(모뎀포함)은 특별히 Windows에서 동작하도록 설계되고 Linux에서 쓸수 없는 전문적인 Microsoft Windows조작체계의 문서고쏘프트웨어를 리용하여 구성된다. 만일 이런 기판이 있으면 보통 다음과 같은 4가지 선택항목들을 가진다.

- Windows 95나 98로 모뎀을 구성한다. 그다음 Linux로 기동한다.일부 모뎀들은 자기들의 구성을 보존한다.
- 모뎀에 대한 Linux구동프로그람을 찾을수 있어야 한다. 일부 제작자들은 Linux구동프로그람을 플로피디스크에 보관한다. 한편 다른것들은 그것을 자기들의 Web싸이트에 가질수 있다.
- 일부 모뎀제작자들은 모뎀의 IRQ, I/O, DMA(만일 필요하면)를 설정하기 위해서 리용할수 있는 구성플로피디스크(보통 MS-DOS방식으로)를 포함한다.
- 이 선택항목들중 아무것도 동작하지 않으면(그리고 모뎀제작자들이 다른 대안을 제공하지 않으면) 모뎀은 콤퓨터가 강제순환될 때마다 구성되여야 하며 Windows는 기동과정에 이것을 진행하지 않으면 안된다. 이런 경우라면 체계를 재기동하지 않고 Linux를 기동하는 loadlin을 리용하여 Windows으로부터 Linux를 기동해야 한다. Loadlin에 대한 보다 구체적인 정보에 대해서는 http://www.linuxdoc.org/HOWTO/mini/Loadlin+Win95-98-ME. html에 있는 The Loadlin+Win95/98/ME miniHOWTO를 참고해야 한다.

여전히 모뎀을 구성하는 문제가 생기면 외장모뎀을 고려해야 한다. 그것이 취급된 포구(보통 $/dev/ttyS_0$나 $/dev/ttyS_1$)가 이미 구성되였기때문에 구성하기 아주 쉽다

주의	쏘프트웨어모뎀으로 알려 진 Windows계 모뎀이 있으면 Microsoft Windows 조작체계쏘프트웨어로 작업하기 위해서 설계된 모뎀을 가진다. Linux는 www.linmodems.org에서 본것처럼 많은 Windows계모뎀으로 작업한다.

모뎀으로 작업하는 쏘프트웨어

Linux에는 모뎀과 함께 쓰이는 많은 류형의 쏘프트웨어들이 있다. 여기에는 다음과 같은것들이 속한다.

minicom 본문식(기본)말단모방프로그람묶음

seyon X Windows식말단모방프로그람묶음

pppd 인터네트에로의 PPP접속을 확립하기 위한 데몬(제22장에서 본다.)

efax 팍스들을 송수신하는데 리용되는 프로그람의 집합(제25장 《Linux에 의한 팍스통신》에서 본다.)

전화접속방식인터네트를 접속하기 위한 일부 GUI편의프로그람들은 모뎀들도 검출할수 있다. 이것들은 제22장에서 서술하게 된다. 다음절은 모뎀이 정확히 작업하고 있는가를 확인하기 위해 minicom리용방법을 속성으로 보여 준다.

모뎀을 검사하는 minicom의 리용

minicom리용의 첫 단계는 대역구성파일을 작성하는것이다. 이것은 뿌리사용자로서 –s기발과 함께 minicom을 실행하면 된다. 즉

```
# minicom –s
```

이 지령은 그림 18-10에서 보여 준것처럼 minicom구성환경을 전개한다.

그림 18-10. minicom구성

사용자는 웃방향건과 아래방향건을 리용하여 차림표들을 이동시키고 Enter건을 눌러서 안내항목들을 선택할수 있다.

모뎀을 검사하기 위한 중요한 설정들은 Serial PortSetup항목에 있다. 이 항목을 선택하면 그림 18-11에서 보여 준 직렬포구설정화면이 현시된다.

```
A -    Serial Device      : /dev/modem
B - Lockfile Location     : /var/lock
C -   Callin Program      :
D -  Callout Program      :
E -    Bps/Par/Bits       : 38400 8N1
F - Hardware Flow Control : Yes
G - Software Flow Control : No

   Change which setting? []

        Screen and keyboard
        Save setup as dfl
        Save setup as..
        Exit
        Exit from Minicom
```

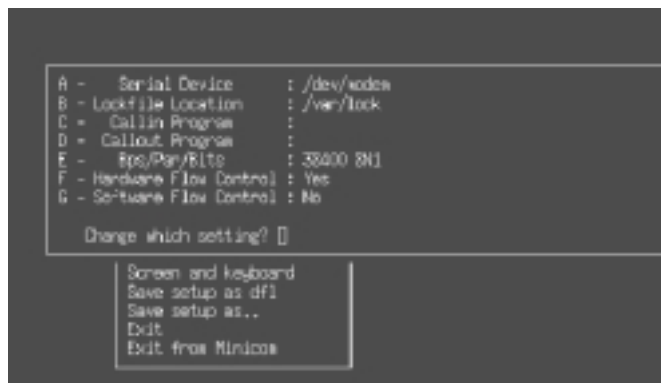



그림 18-11. 직렬포구구성

설정값들을 변화시키려면 그것들을 문자로 선택해야 한다. 먼저 사용자는 직렬장치를 설정해야 한다. 이것은 항목 A 즉 Serial Device로 진행한다. 사용자는 해당한 장치파일로 이 위치를 변화시켜야 한다. 실례로 사용자모뎀이 두번째 직렬포구에 있다면 외부접속에 대하여 minicom을 리용하고 있을 때 /dev/ttyS$_1$로 이것을 설정할수 있다.

일러두기 만일 직렬포구가 그림 18-11에서 보여 준것처럼 /dev/modem에 이미 설정되여 있으면 ls -l /dev/modem지령으로 그 접속을 검색해야 한다. 만일 그것이 직렬포구와 정확히 련결되여 있다면 변화시킬 필요가 없다.

사용자는 또한 사용자모뎀의 설정과 련결하려고 하는 체계에 필요한 설정들을 맞추기 위하여 항목 E 즉 Bps/Par/Bits를 구성해야 한다(보통 인터네트공급자의 체계). 항목 E는 그림 18-12에서 보여 준 통신파라메터화면을 현시한다.

```
                    [Comm Parameters]
A -    Serial   | Current: 38400 8N1
B - Lockfile L  |
C -   Callin P  |  Speed          Parity         Data
D -  Callout P  |
E -    Bps/Par  | A: 300         L: None        S: 5
F - Hardware F  | B: 1200        M: Even        T: 6
G - Software F  | C: 2400        N: Odd         U: 7
                | D: 4800        O: Mark        V: 8
  Change whic   | E: 9600        P: Space
                | F: 19200                     Stopbits
     Screen     | G: 38400                      W: 1
     Save s     | H: 57600                      X: 2
     Save s     | I: 115200      Q: 8-N-1
     Exit       | J: 230400      R: 7-E-1
     Exit f     |
                | Choice, or <Enter> to exit? []
```

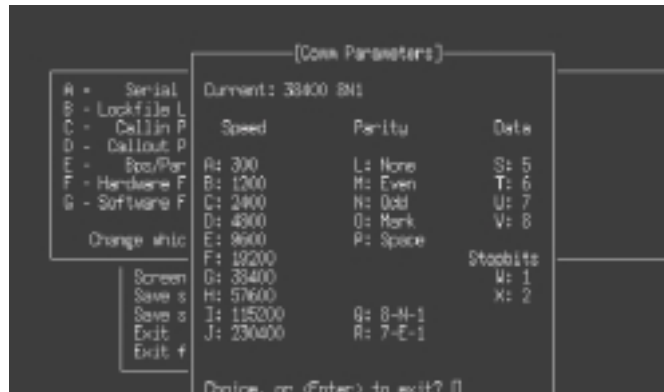



그림 18-12. 통신파라메터의 구성

화면에서 현재설정을 보여 준다. 사용자는 해당한 문자를 선택하여 설정을 변경시킬 수 있다.

고속으로 선택하려면 사용자는 자기의 모뎀에 있는 압축된 접속속도를 선택해야 한다. 이 속도는 일반적으로 사용자모뎀속도의 4배이다. 실례로 14.4Kbps모뎀에서는 57600bps이고 28.8Kbps모뎀에서는 115200bps로 설정된다. 가능하다면 56Kbps모뎀은 115200bps 또는 230400bps로 선택될수 있다.

오늘날 기우성 및 자료비트에 대하여 대부분 접속들의 기준은 기우성이 없는 8개의 자료비트와 1개의 정지비트이다(선택항목 Q). 정확한 설정을 결정하려면 사용자가 접속하고 있는 체계의 관리기와 토의해야 한다.

이것들이 완료되면 직렬포구설정화면으로 돌아 가기 위해 Enter건을 눌러야 한다. 직렬포구설정화면에서 Enter를 누르면 기본차림표에로 넘어 간다.

이 정보는 사용자가 모뎀을 검사하는데 충분해야 한다. 기정설정과 같은 구성을 보관하기 위하여 Save Setup As Dfl 을 선택한 다음 설정화면을 끝내기 위해 Exit를 선택하고 선택한 설정들로 minicom을 실행해야 한다. 이것은 그림 18-13에서 보여 준것처럼 minicom말단모방기를 현시한다.

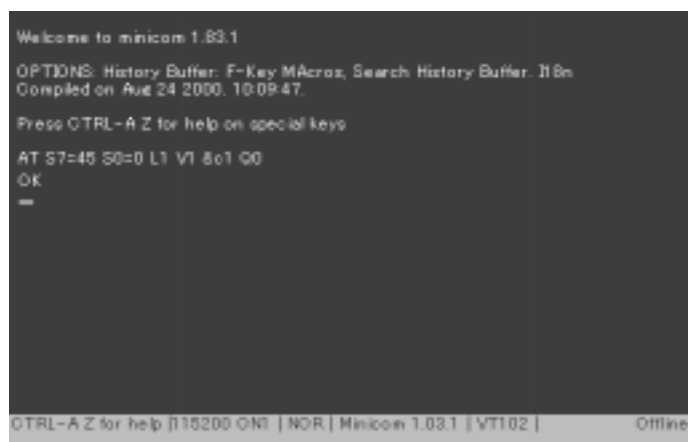



그림 18-13. 기본minicom화면

OK재촉문이 나타나면 모뎀은 정확히 구성되여 작업한다. 사용자는 AT를 입력하고 Enter건을 눌러 이것을 더 검사할수 있다. 그러면 OK통보문을 얻을수 있다. 즉

```
AT
OK
```

만일 이것이 작업하면 접속할수 있는가를 알아 보기 위해 체계에로 다음과 같은 전화번호를 돌리려고 할수 있다.

```
ATDT1234567
```

만일 이것이 작업하면 접속소리를 들을수 있고 접속통보문과 정확히는 원격체계에 대한 일부 통보문이나 재촉문을 보게 된다.

```
ATDT1234567
CONNECT 115200
```

만일 이 단계중의 하나라도 문제가 생기면 사용자모뎀의 물리적접속이나 구성은 정확하지 못할수 있다. 이 경우에 Linux에서 작업하는 모뎀을 얻기 위한 상세한 정보들을 보려면 http://www.linuxdoc.org/HOWTO/ Modem-HOWTO.html에 있는 Modem HOWTO항목을 참고해야 한다.

ISDN정합기에서 제기되는 문제

ISDN 정합기에서는 Linux에서 작업할 때 약간의 문제들이 생길수 있다. 이 어려운 문제는 외부정합기와 내부정합기의 기능이 기본적으로 방법들을 달리 한다는 사실로부터 생기며 내부정합기가운데는 분명히 각이한 기술적인 수법이 있다. 표준ISDN접속은 전용으로 128Kbps 혹은 144Kbps로 제한되여 있고 국제전화표준에 따른다.

외부ISDN정합기들은 Linux에서 그것들의 리용방법을 알기 쉽게 하며 외부모뎀의 동작을 모방하기때문에 Linux체계에서 가장 좋은것을 선택할수 있다.

그러나 내부정합기에서는 문제들이 제기된다. 일부 내부정합기들은 표준직렬모뎀들을 모방하며 다른것들은 표준망기판을 모방하는데 이것은 둘다 Linux에서 작업할수 있다. 그러나 알려 지지 않은 작업도 있다. 내부ISDN정합기들에서 생기는 문제들을 처리하려면 http://www.isdn4linux.de/faq/에 있는 isdn4linux FAQ를 참고해야 한다. Red Hat의 ISDN구성도구인 isdn-config를 비롯하여 여러개의 ISDN구성편의프로그람들을 리용할수 있다. 사용자는 지령행으로부터 직접 이 도구를 실행하거나 일반적인 Red Hat 의 internet-config도구를 통하여 이 도구를 실행할수 있다. GNOME에서 이 도구를 실행하려면 Main Menu단추를 찰칵하고 그다음에 Programs → System → Internet-config를 선택해야 한다. 그림 18-14에서 보여 준것과 같이 나타나는 차림표에서 전화접속방식으로 ISDN을 선택하고 Next를 찰칵한다.

Linux와 ISDN에 대한 상세한 정보를 보려면 http://www.muc.de/hm/linux/linux-isdn. html에 있는 ISDN Solutions for Linux페지를 참고해야 한다.

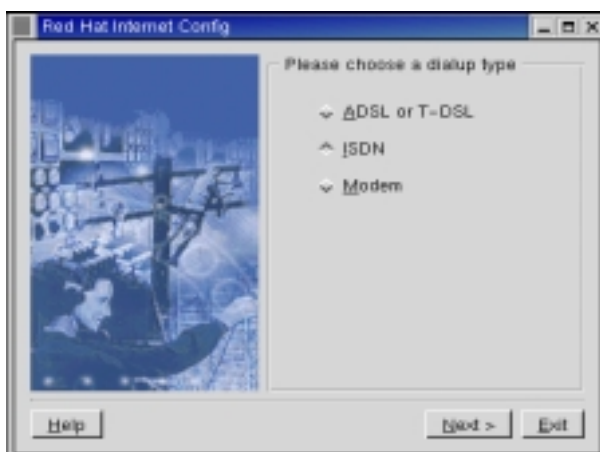



그림 18-14. internet-config에서 ISDN접속을 설정

기타 《모뎀》

오늘날 ISDN보다 더 빠른 봉사기들이 많다. 이전에는 이것들을 광대역봉사기라고도 했다. 사용자는 이미 전화선을 리용한 TV케블과 Digital SubScriber Lines(DSL)을 통한 방송대역접속이나 위성을 통한 접속, 고속무선접속을 가질수 있다.

이 봉사기들을 지원하는 많은 회사들은《모뎀》이라는 봉사기도 준다. 케블모뎀이라고 하는 DSL모뎀들과 위성모뎀들은 오늘날 기본으로 되고 있다. 기술적으로는 이것이 불안하다. 이러한 《모뎀》들에서는 인터네트에 접속하는 minicom, seyon, pppd, KPPP, Microsoft Dial-up Networking을 리용할수 없다. 이 장치들은 더 정확히는 정합기라고 한다.

일반적으로 이 정합기들은 망기판으로 콤퓨터에 접속된다. 만일 사용자가 이 정합기중의 하나를 구성하려고 한다면 제6편 《소규모사무/가정사무에서 Linux의 리용》과 특히 제28장 《이써네트망을 위한 Red Hat Linux 7.1의 구성》을 참고해야 한다.

USB마우스 및 USB건반

가장 최신인 대부분 Linux배포물은 USB마우스와 건반을 지원한다. 이러한 대부분의 지원들은 Linux 핵심부 2.4가 공개되기전에 여러가지 Linux배포물에 포함되여 있었다. 이 두가지 입력장치들은 제12장 《고급한 X Windows구성》에서 보았다. 또한 이것들은 다 USB Human Interface Device나 HID로 정의된다.

주의 판본 2.2.7로 시작하는 Linux 핵심부들은 USB장치들을 망라하기 시작하였다. 현실성 있는 USB을 실행하려면 판본 2.2.18이나 2.4.2및 그이상의 핵심부판본을 가져야 한다.

대부분의 경우에 Red Hat Linux 7.1설치프로그람은 자기의 USB마우스 또는(그리고) 건반을 자동적으로 검출한다. 그러나 아직 문제점들이 생기므로 USB가 마우스와 건

반을 지원하는 방법에 대한 약간의 정보가 필요하다.

첫 단계는 dmesg | less지령으로 핵심부의 통보문들을 검색하는것이다. 적재된 USB나 USB장치들을 보려면 통보문을 훑어 보아야 한다. 만일 사용자가 필요한 통보문을 검색할수 없다면 /proc/bus/usb/devices 파일에 대한 변경들을 검색해야 한다. 사용자는 ls –l /proc/bus/usb/devices지령으로 마지막으로 변경된 날자와 시간을 볼수 있다.

만일 사용자가 자기의 USB하드웨어에 대한 아무런 응답도 받지 못하면 그에 필요한 코드가 핵심부의 부분으로 작성되여 있지 않을수 있다. 사용자는 자기의 핵심부에 그것을 콤파일해 넣을수도 있고 적당한 모듈에 추가할수도 있다. USB에서는 다음과 같은 지령으로 USB장치파일체계를 설치해야 한다(뿌리사용자라면).

```
# mount -t usbdevfs none /proc/bus/usb
```

USB HID에서는 Input core support항목에 마우스와 건반지원은 물론 핵심부에 USB Human Interface Device지원을 다시 콤파일해 넣어야 한다. 한편 사용자는 input.o, hid.o, mousedev.o, keybddev.o모듈을 추가할수 있다. 사용자는 insmod모듈지령으로 한개 모듈을 추가할수 있다. 핵심부에 모듈을 재콤파일하거나 갱신하거나 추가하는 방법에 대한 상세한 정보는 제20장을 참고하면 된다.

USB마우스, 건반 등에 대한 더 상세한 정보는 Linux USB Project의 부분인 Red Hardel문서 《Linux USB보조체계》에 있다. 사용자는 http://www.linux-usb.org/USB-guide/book1.html에서 이 문서를 직접 읽어 볼수 있다.

USB 마우스

모든것이 정확히 설치되면 USB는 다중마우스를 지원한다. Linux는 하나의 마우스구동프로그람 /dev/input/mice에 모든 USB마우스를 구성할수 있다. 만일 앞에서 본 단계들에서 Linux배포물이 마우스 또는 USB마우스를 검출하지 않으면 자기의 Linux핵심부에 필요한 변경들을 가해야 한다. 이 과정을 모듈들을 추가하여 진행하려면 mousedev.o모듈도 리용할수 있는가 확인해 보아야 한다.

필요한 마우스구동프로그람이 없으면 그것을 작성해야 한다. 필요하다면 /dev/input 등록부를 만든 다음 다음의 지령으로 USB부분장치를 만들어야 한다.

```
# mknod /dev/input/mice c 13 63
```

USB마우스나 X Window에 있는 마우스들을 리용하려고 한다고 해도 역시 련관된 구성파일 XF86Config에 USB 마우스나 마우스들에 대한 구성자료를 추가해야 한다. 배치와 기초적인 문법은 제12장에서 고찰하였다. 표준USB마우스에 대하여 다음의 InputDevice항목을 추가해야 한다.

```
Section "InputDevice"
    Identifier   "USB Mice"
    Driver       "mouse"
    Option       "Protocol"   "IMPS/2"
    Option       "Device"     "/dev/input/mice"
EndSection
```

또한 사용자는 화면에 이러한 새로운 입력장치를 결합해야 한다. 제12장에서 본것처럼 이것은 XF86config파일의 ServerLayout에서 진행된다. 만일 이 USB마우스가 유일한 마우스라면 그것은 다음과 같은 ServerLayout항목에 행으로 추가할수 있는 핵심지시자이다.

InputDevice "USB Mice" "CorePointer"

만일 일상적인 마우스(USB가 아닌)도 리용하고 있다면 이것은 다음과 같이 두번째 Input Device행으로 된다.

InputDevice "USB Mice" "SendcoreEvents"

이것은 XFree86 판본4.0에서 기초하였기때문에 이것들의 문법에 추가되는것들은 배포물과(혹은) X Windows의 판본에 따라 각이할수 있다.

USB건반

더 많은 새로운 콤퓨터들은 어미기판들에 USB뿌리하브를 통합한다. 결합된 BIOS는 USB건반을 정상적으로 지원한다. 이 경우에 대체로 Linux핵심부에는 건반을 지원하는 USB정보가 필요 없다.

그러나 만일 자기의 콤퓨터에 Unix/Linux조작체계가 아닌 또 다른 조작체계가 있고 USB를 구성하지 않는 BIOS나 특수한 특징을 가진 건반을 리용하고 있다면 사용자는 USB건반을 제공해야 한다. 만일 이것을 진행하기 위해서 핵심부를 콤파일해 넣으려면 USB HIDBP건반지원을 하지 말아야 한다. 왜냐하면 이것은 Linux에서 대부분의 작업을 지원하지 않기때문이다.

경고	Linux Loader 즉 LILO는 Linux가 임의의 USB건반을 검출하기전에 실행된다. 그러므로 USB지원기능이 콤퓨터주기판에 포함되여 있지 않고 BIOS로 검출되지 않는 한 LILO boot:재촉문에 건반으로 입력할수 없다. 이 문제점들을 해결하기 위하여 Partition Magic나 System Commander와 같은 다른 기동적재기를 리용하는것을 고려해 보아야 한다.

▶미리 알아두기

제19장 《Linux 다매체》에서는 대부분의 사용자들이 중요하게 찾는 분야들에 대하여 배우게 된다. 음성기판과 같은 공통적인 다매체주변장치들을 설치하고 구성하는 방법을 보게 된다. 그리고 공통적인 Linux다매체쏘프트웨어를 통하여 빨리 그것들을 다지게 된다.

제19장. Linux다매체

이 장에서는 Windows와 Mac OS의 다매체와 같은 경쟁적인 가동환경에서와 같이 아직도 미개척분야인 Linux의 분야를 고찰한다. 여기서는 음성기판에 대한 Red Hat Linux 7.1의 지원기능과 그 기판을 구성하는 방법을 주었다. 다음에 전형적인 다매체응용프로그람인 XPlaycd를 서술하고 리용가능한 기타 다매체응용프로그람들을 개괄한다.

Linux다매체를 완전히 파악한다는것은 좀 어려운 일이다. 하드웨어의 지원기능에 대한 정보는 http://www.linuxdoc.org/HOWTO/Sound- HOWTO.html의 Linux Sound HOWTO에서 볼수 있다. 다매체응용프로그람지원기능 즉 Linux에서 음성작업방법에 대해서는 http://www.linuxdoc.prg/HOWTO/Sound-Playing-HOWTO.ht의 Linux Sound Playing HOWTO를 참고할수 있다.

음성기판구성

일반적으로 음성기판에 대한 Linux의 지원기능은 미완성으로 남아 있다. 강력한 지원기능은 Sound Blaster기판과 하드웨어-등록기준위에서 Sound Blaster기판과 호환할수 있는 기판에 제공된다.《호환할수 있다》고 볼수 있는 기판들은 표준Sound Blaster구동프로그람으로 작업하지 않는다.

음성기판에 대한 기타 구동프로그람들은 Red Hat Linux 7.1 sndconfig 편의프로그람으로 구성될수 있지만 그것들 모두는 구성에서 자체의 고유한 특성과 특수한 방법을 가지므로 이 절에서는 Sound Blaster준위의 기판들만을 취급한다. 우에서 언급된 Linux Sound HOWTO 는 Linux체계에서 자기가 직접 대면하게 될 기판을 찾는 방법을 주는 좋은 참고서이다.

앞으로는 음성기판의 방송령역에 대한 지원기능의 준위가 개선되고 Red Hat Linux 7.1을 비롯하여 여러가지 현대적인 Linux배포물들은 음성기판들을 많이 선택할수 있는 기능을 지원하겠지만 이것은 Windows 98에 있는 음성지원기능만큼은 완성되지 못하고 있다.

음성지원의 추가적인 기능은 핵심부의 원천코드를 재콤파일해야 한다는 뜻을 담고 있다. 이것은 제20장 《Linux핵심부의 재콤파일》에서 보게 되는 어려운 과제이다. 핵심부를 콤파일하는데서 정확한 결심을 내리는것은 어려우며 결심을 잘못하면 체계가 작업상태로 돌아 가기 어렵고 기동할수 없게 된다.

이러한 리유로 해서 Red Hat가 적재할수 있는 모듈단위로 Sound Blaster구동프로그람을 개발하였기때문에 핵심부를 다시 구성하지 않아도 Red Hat Linux 7.1체계에 Sound Blaster지원기능을 추가할수 있다.

주의 기판을 구성하려면 기판에 대한 I/O 포구, IRQ, DMA주소와 같은 모든 하드웨어설정기능들을 알아야 한다. 기판설정기능들을 결정하는 지령들에 대해서는 기판문서를 참고하면 된다.

Sndconfig의 리용

Red Hat에는 Sound Blaster모듈을 구성하기 위한 sndconfig프로그람이 포함되여 있다. sndconfig프로그람을 실행하려면 뿌리방식으로 넘어 가서 조종탁이나 Xterm창문의 지령재촉문에 다음과 같은 지령을 입력해야 한다.

```
# /sbin/sndconfig
```

그림 19-1에 보여 준 초기화면은 sndconfig가 PnP음성기판들에 대해서 체계를 조사하는 과정을 보여 준다. 만일 음성기판을 검출했다면 그 기판의 IRQ포구, I/O주소, DMA통로는 자동적으로 기록된다. 그렇지 않으면 기판을 검출하지 못했다는것을 알려준다. OK를 찰칵하면 다음화면은 그림 19-2에서 보여 준것과 같이 사용자의 음성기판류형을 식별할것을 요구한다.

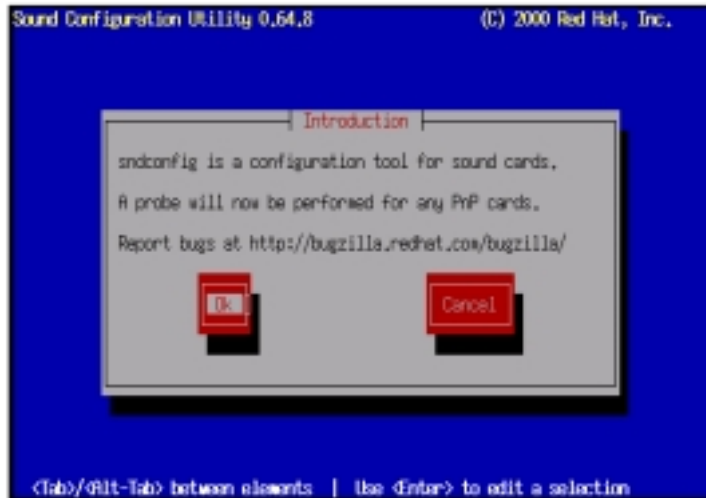



그림 19-1. sndconfig화면

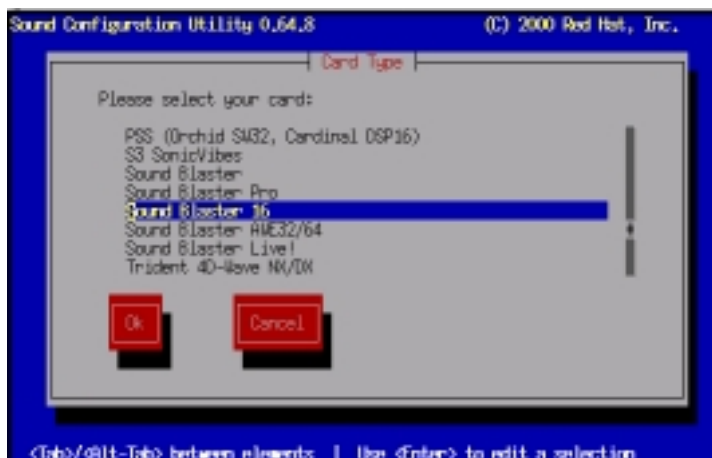



그림 19-2. Sound Blaster기판의 선택

기판류형은 표준으로 지원된 기판목록으로부터 선택할수 있다. 만일 사용자가 정확한 하드웨어호환기판을 가지고 있다면 그것에 맞는 기판류형을 선택해야 한다.

다음에 나타나는 화면은 선택된 기판류형에 따른다. 그 화면은 기판에 대한 I/O포구, IRQ, DMA주소와 같은 정확한 하드웨어의 설정기능들을 제공할것을 요구한다. 그림 19-3은 정규적인 Sound Blaster 16기판에 대한 화면을 보여 준다.

그림 19-3. Sound Blaster 16기판에 대한 하드웨어설정기능들을 정의

일단 이 설정들을 정의하고 확인했다면 sndconfig는 /etc/modules.conf파일에 이 내용을 추가할수 있다(그리고 초기파일을 modules.conf.bak로 여벌복사한다.). 사용자가 OK를 누르면 sndconfig는 검사음성을 설정한다. 다시 OK를 누르면 그 검사음성이 재생된다(회화체의 소리). 다음화면은 그 소리를 들었는가를 확인할것을 요구한다.

주의	Linux의 낡은 판본(Red Hat Linux 7.0으로 배포되기전)에서는 /etc/modules.conf파일대신에 /etc/conf.modules파일을 리용하였다.

다음에 sunconfig는 그림 19-4에서 보여 준것과 같은 검사 MIDI음성화면을 현시한다.

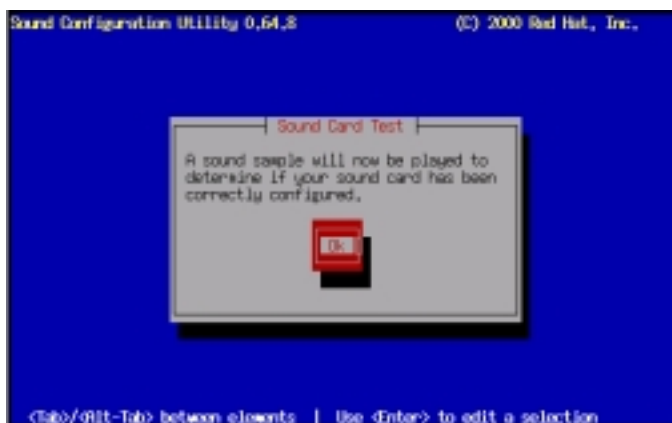



그림 19-4. MIDI음성검사

소리가 나지 않는 경우

만일 sndconfig가 음성재생을 실패하면 어느 하나가 잘못 설정되였을수 있다. 우선 음성기판이 PC에 정확히 삽입되였는가를 검색하고 스피카가 기판에 정확히 끼워 졌는가를 검색한 다음스위치를 넣어 전원이 들어 오는가를 검색해야 한다.

만일 이러한 물리적인 설정들이 제대로 되여 있다면 두가지 어려운 문제가 제기된다. 즉

- 사용자가 기판류형을 잘못 선택했거나 잘못된 하드웨어설정기능들을 제공하였다. 이 모든것들을 구체적으로 검사하고 sndconfig를 다시 실행해야 한다.
- 이 기판은 sndconfig편의프로그람에서 본 음성기판중에는 제공되지 않았다. Linux음성에 대한 더 상세한 정보에 대해서는 Linux Sound HOWTO를 참고하며 자기기판에 대한 정확한 구동프로그람으로 새로운 핵심부를 만들어야 한다.

XPlaycd의 리용

음성기판들에 대한 통속적인 한가지 리용방법은 음성기판의 스피카를 통하여 음성 CD-ROM들을 재생하는것이다. 이것은 XPlaycd프로그람으로 실행된다. 이것은 Red Hat Linux 7.1에 포함되여 있고 CD-ROM구동기에 간단한 도형대면부를 제공하는 X Windows응용프로그람이다. 사용자는 GNOME Main Menu단추에 있는 Programs → Multimedia → XPlayCD를 선택하여 이 프로그람을 호출할수 있다.

그림 19-5에서 보여 준것과 같이 XPlaycd는 누구에게나 립체음성CD재생도구로 잘 알려 져 있다.

우에 놓인 왼쪽에서부터 오른쪽까지의 조종단추들은 다음기능을 수행한다.

Exit XPlaycd를 완료한다.

Open/Close 쏘프트웨어로 조종되는 투사를 제공하는 구동기들에서 CD판을 교대로 내보내고 닫는다.

Rewind 2초간격으로 현재자리길을 거꾸로 주사한다.

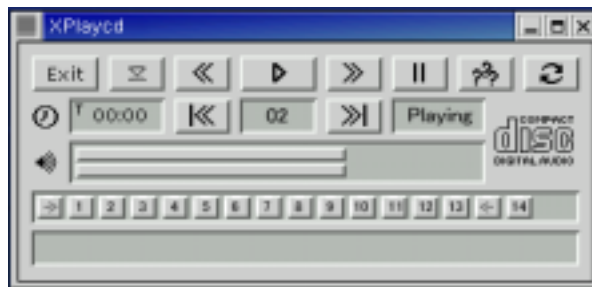



그림 19-5. Xplaycd창문

Play/Stop 현재자리길을 교대적으로 재생하거나 정지한다.

Fast Forward 2초간격으로 현재자리길을 앞으로 주사한다.

Pause/Continue 재생을 순간정지하거나 계속한다.

Shuffle/Resort 불규칙적으로 자리길을 재생하거나 물리적인 순서로 복귀시킨다.

Repeat 재생목록을 계속 반복한다.

Rewind단추의 바로 아래에는 Back Track단추가 있다. 일단 Back Track단추를 누르면 현재자리길의 시작으로 넘어 가며 그것을 다시 누르면 한번에 한 자리길씩 거꾸로 이동한다. Fast Forward단추의 바로 아래에는 Forward Track단추가 있는데 이것을 누르면 다음번 자리길의 시작으로 넘어 간다.

조종단추의 세번째 렬에는 음향조절기능이 있다. 여기에서 매 통로를 각이하게 조절

하거나 두 통로의 값을 동시에 변화시킬수 있다. 음향띠의 설정을 변화시키려면 요구되는 방향 즉 왼쪽 또는 오른쪽으로 띠를 한눈금씩 이동하여 현재설정을 바꾸어야 한다. 둘중 어느한 방향으로 두 띠들을 이동하려면 해당한 쪽을 찰칵해야 하지만 실제로는 두 띠사이를 찰칵해도 된다.

마지막으로 음향조절띠아래에는 재생목록이 있다. 재생목록은 수자들의 순서로 놓인 매 자리길에 대한 작은 아이콘을 표시한다. 시작아이콘은 오른쪽화살표식아이콘으로 표시되며 끝아이콘은 왼쪽화살표식아이콘으로 표시된다.

재생목록은 시작아이콘으로부터 끝아이콘까지 재생되며 재생목록의 왼쪽으로부터 오른쪽으로 가면서 두 아이콘에사이에 있는 매 자리길아이콘들이 재생된다. 사용자는 재생목록우에서 임의의 아이콘의 위치를 변화시키기 위하여 그 아이콘을 찰칵하여 오른쪽 혹은 왼쪽으로 끌어 갈수 있다. 그림 19-6은 재생목록의 모든 자리길들이 교대로 재생되는 것을 보여 주지만 순서는 불규칙적으로 되여 있다.

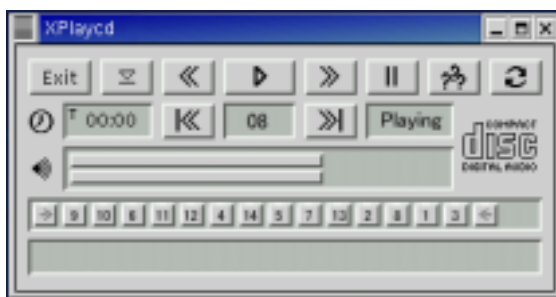



그림 19-6. 불규칙적인 재생목록

그림 19-7에서 보여 주는것처럼 시작아이콘 및 끝아이콘들을 이동하여 자리길의 일부를 삭제한 재생목록을 만들수도 있다.

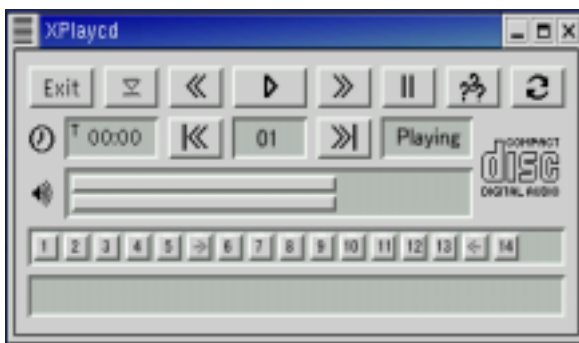



그림 19-7. 재생목록의 시작아이콘 및 끝아이콘의 조정

기타 다매체응용프로그람

기능이 잘 째이고 다양한 다매체응용프로그람과 유희도구는 다른 가동환경에서 쓸수 없었지만 Linux에서는 쓸수 있게 많은 응용프로그람을 개발되였다.

http://www.linuxapps.com 의 Linux Apps페지에서 Linux 에 대한 많은 쓸모 있

는 음성, 동화상 그리고 기타 다매체응용프로그람들의 목록을 볼수 있다. 이 절에서는 간단히 선택할수 있는 다매체응용프로그람들을 잠간 고찰한다.

GNOME의 CD재생프로그람

Red Hat Linux 7.1에서 기정탁상관리기인 GNOME은 그림 19-8에서 보여 준것과 같은 함축된 CD재생프로그람을 가지고 있다. GNOME Main Menu의 Programs→ Multimedia →CD Player를 선택하여 이 프로그람을 호출할수 있다.

그림 19-8. GNOME의 CD재생프로그람

이 CD재생프로그람은 요구되는 표준기능을 제공하는데 이 기능은 대부분의 물리적인 CD재생장치들의 조종단추는 물론 대부분의 PC에 기초한 CD재생장치와 비슷한 간단한 도형조종단추들을 집중시킨다. 특히 이 응용프로그람은 재생하려는 자리길의 순서를 선정하는 견고한 자리길편집도구도 가지고 있으며 일부 Linux에 기초한 CD재생프로그람들에 없는 건반조종과 같은 기타 특징들도 제공한다.

MpegTV 재생프로그람1.0

일반적으로 미국화페로 10딸라인 이 응용프로그람그람은 Linux와 기타 Unix가동환경에 대한 실시간MPEG비데오/음향재생기이다. 그것은 파일들의 재생을 조종할수 있고 한개의 열린 파일의 임의의 위치에로 넘어 갈수 있으며 음향을 조정할수 있는 직관적인 조종판과 같은 간단한것을 제공한다. 쏘프트웨어는 http://www.mpegtv.com/에서 내리적재할수 있다.

Festival음성합성체계

Festival음성합성체계는 강력하면서도 각이한 언어(영어, 에스빠냐어, 웨일즈어)로 된 대화합성체계를 제공하도록 설계되였다. 이 체계는 응용프로그람개발자들에게 자기의 쏘프트웨어로 대화를 추가하는데 필요한 도구들을 제공하며 본문을 대화식으로 바꾸는데 쓰일수 있다.이 프로그람묶음의 현행공개판본은 http://www.cstr.ed.ac.uk/projects/festival/festival.html에서 리용할수 있다.

MiXViews

MiXViews는 무료로 배포할수 있는 X Windows용수자음성편집기이다. 자료에 파일들의 자르기와 붙이기 그리고 음성변경을 적용하도록 하면서 다중파일들을 현시 및 편집할수 있다. MiXViews홈페지는 http://www.create.ucsb. edu/˜doug/htm1s/MiXViews.htm1에 있다.

Grio와 Krio

휴대용기억매체를 리용하는 음악재생프로그람들이 발전함에 따라 MPEG 부호기 또는 복호기들에 대한 수요가 증가한다. GNOME과 KDE탁상을 위한 응용프로그람들을 비롯하여 Rio재생프로그람과 같은 체계의 입력관리를 위한 많은 응용프로그람들이 있다. 그것들은 grio와 krio로 명명되였다.

이 두 응용프로그람들은 재생순서를 다시 배정하며 재생프로그람을 초기화하고 필요에 따라 파일들을 지우고 Rio재생프로그람으로부터 혹은 거기에로 자료를 전송하도록 한다. http://kipper.crk.umn.edu/˜gerla/grio/에서 grio를, http://krio.sourceforge.net/index.php에서 krio를 내리적재할수 있다.

RealPlayer

RealNetworks로부터 적재할수 있는 RealPlayer는 인터네트를 통하여 실시간적으로 RealAudio와 RealVideo파일들을 재생하는 인기 있는 재생프로그람쏘프트웨어이다. 대중적인 Web방식라지오, 텔레비존, 음악싸이트들은 RealAudio와 RealVideo기술을 리용하며 RealPlayer는 이 내용을 보거나 듣는데 쓰일수 있다. 재생프로그람의 무료적이며 업무적인 판본들은 http://scopes.real.com/real/player/unix/unix.htm1?src=rpbformdpgi에서 적재할수 있다.

FreePhone

FreePhone은 Windows와 Macintosh체계에서 볼수 있는 인터네트전화쏘프트웨어에 대한 개념과 비슷한 인터네트음성전화회의도구이다. 그것은 1대 1이상의 전화체계 즉 사용자들이 유일방송 혹은 다중방송형식의 음성전화회의를 할수 있는 체계이다. FreePhone은 다양한 대중적인 음성압축체계들을 지원하며 인터네트의 다매체골간(Mbone)을 리용한다. 쏘프트웨어는 http://www.inria.fr/rodeo/fphone/obtain.htm1에서 직접 볼수 있다.

▶미리 알아두기

이 장에서는 다매체의 한 부분에 지나지 않는 항목을 보았다. Linux 환경에서 다매체는 절실히 필요하고(그것은 Windows에서도 같다.) 이에 대해서 일반적인 책들에서는 구체적으로 설명하지 않았다.

만일 지원된 기판이나 실제로 호환할수 있는 기판을 리용하고 있다면 이제는 Red Hat Linux체계로 작업할수 있어야 한다. 사용자는 다매체응용프로그람들을 내리적재하여 검사할수 있고 Linux에서 다매체응용프로그람을 리용하면 좀더 편리해 질수 있다.

제20장《Linux핵심부의 재콤파일》에서는 중요하고도 고급한 문제점을 고찰한다. 핵

심부의 재콤파일은 음성기판과 같은 장치에 지원기능을 추가, 덜 공통적인 망규약에 지원기능을 추가, Linux를 설치할 때 설치되는 기정핵심부의 크기줄이기를 비롯하여 많은 전제들을 위해서 요구될수 있다. 여기서는 재콤파일과정에 대해서 배우게 되며 이 과정에 작성해야 할 기본선택기능들에 대해서 고찰하게 된다.

제20장. Linux핵심부의 재콤파일

Linux는 콤퓨터가 작업할 때 맞다들릴수 있는 조작체계의 하나이다. Linux가 조작체계의 핵심부에 대한 모든 원천코드를 가지고 있기때문에 사용자는 마음대로 Linux의 기능을 바꿀수 있다. 물론 그것은 사용자들이 C언어 원천코드파일을 변경하지 않고 제공된 도구로 핵심부를 재구성한다는것을 의미한다.

그럼에로 불과하고 수백개의 선택항목들은 표준핵심부의 구성편의프로그람 부분으로 쓰일수 있다. 이 장에서는 실행하는 Linux체계에 포함시키려는 선택항목들을 선택하기 위하여 핵심부구성도구들을 리용하는 방법을 고찰한 다음 이 선택항목들에 기초한 핵심부원천코드를 재콤파일하는 방법을 보게 된다.

핵심부의 갱신리유

만일 Linux가 처음이라면 사용자는 《왜 핵심부를 재콤파일하려고 하는가?》에 대한 의문을 가질수 있다. 그리고 그것이 대단히 어려운 과정이라고 생각할수 있다.

Linux는 2～3년전보다 훨씬 더 사용자들과 친숙해 졌는데 설치과정의 개선, 도형대면부들과 하드웨어지원기능들은 콤퓨터과학전문가가 아닌 사용자들에게 접근할수 있는 Linux를 만들었다. 사실 여기서 보게 되는 핵심부재콤파일과정은 차례구동구성대면부와 핵심부를 재작성하기 위한 간단한 지령행을 리용하는 과정이다.

Linux 에서는 조정을 아주 잘할수 있다. 즉 사용자는 자기의 일관한 요구가 실현되도록 그의 능력들을 세련시킬수 있다. 그러나 Linux의 많은 특성들은 표준 Linux 핵심부에서 기정으로 처리되지 못한다. 이것은 여러가지 리유들에 따른다.

가장 중요하게 알아야 할것은 매 사용자가 자기들의 쏘프트웨어에 교환조건들을 만드는것이다. 실례로 압축프로그람을 리용하면 사용자는 더 빠른 압축시간을 선택하거나 또는 더 작게 압축된 파일을 선택할수 있다.

같은 방법으로 Linux 핵심부도 교환조건들을 만들어야 한다. 실례로 사용자는 추가적인 구성단계들을 거치지 않고 더 작은 핵심부를 가지거나 하드웨어를 제공하는 핵심부를 가질수 있다. 또는 IP프로그람묶음을 찾아 내는 체계 또는 일반워크스테이션과제들을 조종하는 최량화된 체계를 가질수 있다.

프로그람작성용어 즉 작성, 조립, 번역

핵심부에서 작업하는 과정에 알게 되는것처럼 사용자는 교대적으로 쓰는 3개의 용어들 즉 핵심부를 재작성하기, 재조립하기, 재번역하기를 보게 된다. 사용자는 이것들을 같은 뜻으로 볼수도 있지만 그것은 실제로 매 용어들의 의미를 리해하게 한다.

Linux(그리고 기타 조작체계)에서 쏘프트웨어를 개발하면 원천코드와 많은 문서고들의 쪼각들을 함께 맞추는 방법을 서술하는 프로그람작성대상과제를 위한 구성파일을 리용할수 있다. 이 정보는 make 편의프로그람이 리용하는 Makefile파일에 보관된다(이 장의 마감에 make편의프로그람들을 드문히 리용하게 된다.).

콤퓨터가 리용할수 있는 모든것(2진코드)으로 대상과제의 원천코드를 실제로 바꾸는 과정을 콤파일이라고 한다. Linux에 대한 표준 C언어번역프로그람들인 gcc 또는 egcs는 원천코드를 콤파일하는데 쓰인다.

그러나 일반적으로 프로그람작성대상과제들은 여러가지 각이한 구성요소들로 된 묶음들을 가지고 있기때문에 make편의프로그람은 매 구성요소에 대한 날자와 시간을 검색하고 이미 콤파일된것중에서 갱신된 요소들만 재번역한다. 이것은 원천코드의 부분이 갱신되였을 때에만 대상과제의 새 판본을 작성하는 시간을 보관한다.

make편의프로그람을 리용할 때 사용자는 《scratch로부터》 재번역하려고 하는 대상과제의 모든 구성요소들을 지적할수 있다. 이것을 조립이라고 한다.

핵심부를 다시 작성하고 재조립하고 재번역하는것은 비슷한 과정들로 진행된다. 그리고 여전히 그 과정은 몇십만행의 원천코드로 진행되며 완성될 때까지 그것을 호출하지 말아야 한다.

핵심부는 자기의 크기, 속도, 하드웨어지원사이에 균형을 잘 맞추도록 구성된다. Linux판매자가 의도한것처럼 이 균형은 일반사용자의 요구들을 만족시키려고 한다. 이 장에서는 어느 특징들이 콤퓨터환경에서 가장 중요한가에 대하여 자체로 결정하는 방법을 배우게 된다.

모듈고찰

핵심부의 모듈들은 Linux의 중요한 부분이다. 핵심부의 모듈은 핵심부원천코드를 재콤파일하지 않고도 Linux의 핵에 기능을 추가할수 있다. 이 장의 마지막에서 보게 되는것처럼 이것은 아주 필요한 기능이다.

실례로 만일 특수한 새로운 SCSI대면부기판으로 작업하는데 필요한것을 알아 내면 사용자는 단일지령(insmod지령)으로 그것을 지원하는 핵심부의 모듈을 적재할수 있다.

Linux에 의하여 지원된 하드웨어장치들의 수가 증가함으로써 핵심부의 모듈들은 사용자들이 소유한 하드웨어에 대한 지원을 쉽게 추가하도록 하는 한편 상대적으로 Linux 핵심부가 작아지도록 한다.

일러두기 Linux와 다른 조작체계사이의 큰 차이점은 체계를 매번 재기동하지 않고도 하드웨어, 파일체계, 언어 등에 대한 지원기능을 적재하거나 삭제하는 능력을 가지고 있다는데 있다.

표 20-1에서는 핵심부모듈들에서 리용된 지령들의 묶음을 주었다. 사용자는 이 지령들에 대한 더 자세한 정보를 사용설명서페지에서 볼수 있다.

표 20-1. 핵심부모듈에서 리용된 지령

명 령	설 명
lsmod	실행핵심부에 현재 설치된 모든 모듈들을 표시한다.
insmod	실행핵심부에 모듈을 삽입한다. 이 지령은 모듈이름을 리용한다. 대신에 모듈이 동작하는 방법을 정의하는 추가적인 파라메터들을 리용할수도 있다(실례로 하드웨어장치에 대한 IRQ와 주소정보).
rmmod	실행체계로부터 모듈을 삭제한다. 리용중에 있는 모듈들을 삭제하는 현상이 없도록 주의해야 한다(현재 적재된 모듈들에 대한 상세한 정보는 **lsmod**지령의 출력에서 볼수 있다.).
depmod	한개 모듈묶음을 적재하는 **modprobe** 지령에 리용될수 있는 파일을 작성한다.
modprobe	**depmod**지령에 의해서 정의되는 모듈들의 묶음을 적재한다.

주의 **Lsmod**를 제외한 지목된 모든 지령들은 사용자가 뿌리방식으로 작업할것을 요구한다. Red Hat Linux 7.1에서 이 지령들은 기종경로에 없는 /sbin등록부에 있다. 그러므로 이 지령들 실례로 **/sbin/lsmod, /sbin/insmod**등과 같이 완전한 경로를 입력해야 한다.

Linux체계의 모듈에 기타 정보는 정해 진 체계파일들을 보면 된다.

핵심부모듈들이 실행원천코드로 삽입되기때문에 그것들은 목적코드(이미 번역된)로 보관된다. 매 모듈은 확장자 .o인 파일을 가진다.

다음의 파일들을 검색한다.

- /boot/module-info-2.4.2-2에는 하드웨어의 특수한 설정들을 대신할수 있는 모듈당 정보를 포함된다.
- /lib/modules/2.4.2-2/는 체계에서 쓸수 있는 전체 핵심부의 모듈들을 가르는 여러개의 보조등록부를 가지고 있다. 실례로 여러가지 CD-ROM구동기를 지원할수 있는 모든 모듈들을 보려면 kernel/drivers/cdrom보조등록부의 내용을 검색해야 한다.

주의 Red Hat Linux 7.1과 관련된 핵심부판본은 2.4.2이다. 만일 Linux의 다른 판본을 리용하거나 더 갱신된 핵심부가 있으면 이 판본을 대신할수 있다.

일러두기 모든 Linux 핵심부들은 major, minor, patch형식으로 판본번호들을 가지고 있다. 가장 큰 변경들은 새로운 기본판본번호와 관련된다. minor개정판번호에 대한 변경들은 아주 중요하다. minor개정판이 짝수번호이면 핵심부는 변동이 없는것이라고 볼수 있다. 홀수번호이면 핵심부가 개발단계에 있다는것을 의미하므로 초심자들이나 미완성콤퓨터들에 리용하지 말아야 한다. 증가변경들이 개발됨에 따라 그 변경들은 patch개정판에 포함된다.

재콤파일리유에 대한 리해

만일 Linux 핵심부을 재콤파일하는데 대해서 여전히 의문이 있으면 표 20-2에 있는 정보는 그에 대한 해답을 주게 된다. 이 표에서는 있을수 있는 일반적인 문제들에 대한 해답을 주었다.

물론 이 리유가운데서 한가지만에 대해서라도 핵심부을 재조립해야 한다. 그러나 핵심부로 작업하는 리유를 리해함으로써 Linux의 유연성에 대하여 더 잘 알게 된다.

표 20-2. Linux 핵심부를 재콤파일하는 과정에 제기되는 문제에 대한 해답

문 제	해 답
사용자는 처리프로그람에 체계를 파괴시키는 오유가 있다는것을 알아야 한다. Linux는 낡은 판본의 작업환경에서 작업한다.	Linux핵심부에 대한 림시수정판을 내리적재하고 원천코드를 다시 작성해야 한다. 20분이면 그 문제가 해결된다.
사용자는 Linux핵심부모듈에 지원된 새로운 SCSI기판을 구입했지만 그 SCSI모듈이 기정핵심부의 기정이 아니면 SCSI하드디스크로부터 체계를 기동할수 없다.	필요한 SCSI지원기능을 모듈로 만들지 말고 조립하도록 지적하면서 핵심부를 다시 작성해야 한다.
보안경보는 Linux파일체계구동프로그람에서 제기되는 문제를 나타낸다. 림시수정은 몇시간내에 다시 작성해야 한다.	림시수정판을 내리적재하고 체계에 가장 최신의 보안을 제공하는 핵심부를 다시 작성해야 한다.
새로운 망기판구동프로그람은 체계에서 리용할수 있다. 판매자의 WebF싸이트로부터 천코드를 내리적재한다.	새로운 망기판구동프로그람은 체계에서 망기판모듈과 핵심부를 다시 작성해야 한다.
2.2.16.Linux핵심부를 가지는 업무용 Linux제품을 구입했지만 이제는 최신판본의 개정된 핵심부를 가지고 싶다.	핵심부원천을 내리적재하고 핵심부의 마지막판본을 가지도록 핵심부를 다시 작성해야 한다.
체계상에서 사용자들은 새로운 지령들(실례로 insmod와 같은)을 리용하지 않는한 정확한 파일체계를 호출할수 없다는것을 호소하고 있다.	핵심부모듈을 추가하지 않고 그 형태의 파일체계들이 설치될수 있도록 필요한 파일체계기능을 가진 핵심부를 다시 작성한다.

핵심부갱신판의 조사

Linux 핵심부는 내적으로 매일 변경될수 있다. 다행히도 안정된 체계를 실행할 때 일반적으로 모든 변경들을 따라 가면서 살펴 보지 않아도 된다. 사실 Linux에서 대부분 변경들은 핵심부를 갱신할 때마다 생기는데 그것은 마음대로 제작체계를 실행하려고 하는 변경들은 아니다.

Linux 핵심부와 그와 관련한 문제들로 생기는 갱신판본을 유지하려면 다음과 같은 Web싸이트를 정상적으로 조사해 보아야 한다. 이 싸이트들은 새로운 하루에도 여러번 새로운 정보로 갱신된다.

www.linuxhq.com 이 싸이트는 최신판의 개발제품과 안정한 핵심부들에 대한 항목, 정보자원에로의 련결, Linux 핵심부에 대한 리용가능한 상세한 정보가 들어 있는 종합적인 정보를 가지고 있다.

www.freshmeat.net 이 싸이트는 공개원천쏘프트웨어와 관련된 문제에 대한 공개판들을 통지하기 위한 정보교환소이다. 이 싸이트는 찾을수 있는 공개정보들의 문서고를

가지고 있다. Freshmeat싸이트는 Linux핵심부속도에만 치우치지 않고 핵심부공개판본들에 대한 정보들도 포함한다.

www.linuxtoday.com 이 싸이트는 공개원천쏘프트와 그와 관련된 문제들에 대한 일별소식편람이다. 이 싸이트는 조사할수 있는 공개정보문서고를 포함한다. Linux Today싸이트는 Linux핵심부속도에만 치우치지 않고 새로운 핵심부공개통지문도 포괄하고 있다.

원천프로그람묶음의 검사

Linux 핵심부를 재콤파일하려면 체계상에 설치된 핵심부에 대한 원천코드가 있어야 한다. Red Hat Linux 7.1에서 핵심부원천코드는 기정으로 설치된 한개의 rpm프로그람묶음에 있다. 만일 그것이 설치되지 않았다면 그 프로그람묶음은 다음과 같이 검색해야 한다.

```
kernel -source-2.4.2-2.i386.rpm
```

사용자는 이 책과 함께 제공된 Red Hat 7.1 CD-ROM을 설치한 후에 rpm지령을 리용하여 다음과 같이 이 프로그람묶음을 설치할수 있다.

```
rpm -Uvh kernel -source-2.4.2-2.i386.rpm
```

주의 Red Hat는 파일확장자 .src.rpm인《원천코드프로그람묶음》로서 표기되지 않은 rpm프로그람묶음에 핵심부원천코드를 가지고 있다. .src.rpm프로그람묶음은 사용자에게 필요 없다.
핵심부원천프로그람묶음은 이 책과 함께 제공된 Red Hat Linux 7.1의 Publisher's Edition에는 포함되지 않았지만 http://www.redhat.com으로부터 적재할수 있다.

도구의 검색

자체의 원천코드외에도 원천코드를 재조립할수 있는 콤파일러가 있어야 한다. 필요한 두개 도구들은 다음과 같다.

- make편의프로그람
- C언어콤파일러 gcc 혹은 egcs

이 편의프로그람들에 대한 실제판본들은 체계마다 다를수 있다. gcc2.96이나 egcs 1.1.2.와 같은 콤파일러의 최근판본을 가지는것이 좋다.

Linux체계에서 이 편의프로그람들을 검색하려면 다음과 같은 지령(전형적인 Red Hat를 설치한 후에 출력으로 나타난다.)들을 리용해야 한다.

```
# rpm -q make
make-3.79.1-5
# rpm -q gcc
gcc-2.96-81
```

주의 같은 지령들은 쏘프트웨어프로그람묶음을 위한 **rpm**을 실행하는 Linux 배포물에서 작업해야 한다.

만일 체계에 이러한 도구들이 설치되지 않았다면 더 나가기전에 이 책과 함께 제공된 Red Hat Linux 7.1 Publisher's Edition CD-ROM으로부터 그것들을 설치하는 rpm이나 그와 비슷한 편의프로그람을 리용해야 한다.

일러두기	**gcc**를 설치하려고 하면 **failed dependencies** 통보문을 보게 된다. 이 통보문은 처음에 설치해야 할 기타 **rpm**프로그람묶음을 제시한다. 이 경우에 요구되는 추가적인 프로그람묶음은 CD-ROM에서도 볼수 있다.

경고	만일 현재 핵심부를 위한 기동디스크가 없으면 뿌리사용자방식에서 **/sbin/mkbootdisk -device /dev/fd0 2.4.2-2**지령을 리용하여 기동디스크를 만들수 있다.

낡은 핵심부의 여벌복사

비록 여기서 함축한 Linux 핵심부의 재조립이 아이들의 놀음인것처럼 보이지만 아직은 더 구체적으로 들어 가기전에 조심해야 한다. 그것은 문제가 생기는 경우에 핵심부를 여벌복사하고 핵심부의 이전 판본으로 재기동하는 방법을 제공한다는것을 의미한다.

현재핵심부의 여벌복사는 세 단계를 거쳐서 진행된다.

1. 만일 구성이 잘못 만들어 졌으면 현재 핵심부를 다시 만들어야 하므로 핵심부에 대한 원천코드나무를 여벌복사해야 한다.
2. 이미 알고 있는 동작으로 체계를 기동하기 위한 핵심부가 있으므로 핵심부자체를 여벌복사한다.
3. 핵심부의 여벌복사로부터 시동할수 있으므로 새로운 시동적재기입구를 만들어야 한다.

현재핵심부에 대한 원천코드나무를 여벌복사하기는 간단하다. 바로 다음의 cp지령과 같은 지령을 리용해야 하는데 이 지령은 2중등록부로 전체 핵심부원천코드나무를 복사한다(만일 여벌복사물을 리용해야 하면 초기이름으로 돌아 가서 그것을 복사할수 있다.).

```
# cp -r /usr/src/linux-2.4.2 /usr/src/linux-2.4.2.sav
```

핵심부자체를 여벌복사하는것은 간단하다. 즉 다음의 cp지령과 같은 지령을 리용해야 한다(그것을 리용하고 있는 실제 핵심부판본과 일치하도록 변경한다.)

```
# cp /boot/vmlinuz-2.4.2-2 /boot/vmlinuz-2.4.2-2.orig
```

주의	핵심부를 재조립하면 이미 존재하고 있는 핵심부는 파일확장자 .old로 보관된다. 그러나 낡은 핵심부판본은 체계를 기동하기 위해서 자동적으로 호출할수 없다. 이것이 이 단계들의 문제점이다.

하드구동기로부터 Linux를 정상적으로 기동하는 LILO기동적재기들은 핵심부파일로 지정하여 뿌리파일체계에서 구성한다. 실례로 /etc/lilo.conf파일에서 다음과 같은 행을 볼수 있다.

```
image = /boot/vmlinuz-2.4.2-2
```

이 행은 기동하는 핵심부의 경로와 파일이름을 가리킨다.

일단 핵심부자체의 여벌복사물이 작성되였으면(앞단계에서) 낡은 판본의 Linux체계를 기동할수 있도록 lilo.conf에 다른 항목을 만들어야 한다.

이것을 진행하려면 다음단계들을 거쳐야 한다.

1. 본문편집기에서 /etc/lilo.conf파일을 연다.
2. 핵심부영상을 참고할수 있는 /etc/lilo.conf파일에서 그 항목을 조사한다.
3. 전체 부분(보통 본문의 4행 또는 5행에 대한)을 복사한다.
4. 복사할 때 두 항목을 변경시킨다.
   - 핵심부의 이름을 여벌복사된 핵심부의 이름으로 변경시킨다(실례에서는 .orig로 끝난다.).
   - 표식마당을 linux.original 또는 linux-previous와 같은것으로 변경시킨다.
5. initrd로 시작하는 행이 있으면 initrd-2.4.2-2.orig.img와 같이 이것을 해당한 여벌복사물의 이름으로 변경시킨다. 필요하면 후에 이 파일을 작성하게 된다.
6. 변경된것들을 /etc/lilo.conf에 보관한다.
7. 기동적재기에 새로운 항목들을 추가하기 위하여 지령 /sbin/lilo를 실행한다. 기동적재기에 영상들이 추가되였기때문에 Lilo를 실행하면 화면에서 각이한 표식을 볼수 있다.

다음번에 체계를 재기동하면 도형LILO기동재촉문에 나타나는 새로운 핵심부의 항목을 볼수 있다. 도형LILO기동재촉문을 리용하지 않은 배포물을 쓰고 있다면 핵심부를 재조립하기전에 경계했던 추가적인 영상을 보기 위하여 Tab를 눌러야 한다.

새로운 핵심부의 얻기

새로운 핵심부의 복사물은 여러가지 방법으로 얻을수 있다. 두가지 주요한 방법이 있는데 하나는 http://www.kernel.org에 있는 Linux Kernel Archives를 통하여 얻는것이고 다른 하나는 Linux 배포물의 Web싸이트를 통하여 얻는것이다. 핵심부의 rpm판본을 조사하고 있을 때 중요한 원천은 http://www.rpmfind.net에 배치된다. rpm프로그람묶음의 내리적재하기와 설치하기는 이미 앞의 장들에서 고찰하였다. 그러므로 이 절에서는 linux-2.2.4.tar.gz와 같은 파일에서 tar.gz형태의 핵심부의 내리적재하기를 취급한다.

경고	이 장에서 이미 본것과 같이 핵심부에 대한 이전의 판본을 여벌복사하는것을 확인해야 한다.

일단 요구하는 핵심부가 내리적재되였다면 그것을 /usr/src/등록부에 옮겨야 한다. 이 절에서는 이 장에서 이미 본것과 같은 현재 핵심부를 여벌복사했다고 보아야 한다는것을 명심해야 한다.

핵심부과 그와 련관된 파일들(준위 2.4.4에서 8000개이상)을 추출하려면/usr/src등록부로 넘어 가서 다음의 지령을 실행시켜야 한다.

```
tar zxpvf linux-2.4.4.tar.gz
```

만일 핵심부프로그람묶음들이 각이한 파일이름을 가졌으면 알맞게 바꾸어야 한다. 이제는 새 핵심부를 구성하기 위하여 다음절에서 고찰하게 되는 편의프로그람들가운데서 한개의 프로그람들을 리용할수 있게 되였다.

주의	만일 핵심부를 개량하고 있다면 쏘프트웨어가 upgrade를 조절할만큼 충분히 개선되였는가를 확인해야 한다. 핵심부 2.4계렬에서는 요구되는 중요한 판본에 대하여 /usr/src/linux-2.4.x/Documentation/Changes 파일을 다시 조사하여야 한다.

새로운 핵심부의 구성

새로운 Linux 핵심부를 작성하는데서 흥미 있는 부분은 새로운 핵심부구성를 설정하는것이다. 이것은 핵심부가 어느것을 포함하고 어느것을 포함하지 않는가를 결정하는 단계이다. 새로운 원천으로부터 핵심부를 선택할수도 있고 이미 있는 원천들로부터 선택할수도 있다. 실례로 만일 Red Hat Linux 7.1을 실행하고 있다면 새로운 선택항목으로 이미 있는 2.4.2 핵심부를 다시 구성할수 있다. 대신에 더 새로운 2.4.4 핵심부를 적재하고 설치할수도 있다. 구체적인 구성선택항목들이 달라 져도 도구들과 편의프로그람들, 수법들은 달라 지지 않는다.

Linux는 3가지의 개별적인 구성편의프로그람들을 가진다. 개개의 프로그람들은 여러가지 우선권들에 알맞는다.

지령행대면부 지령행대면부는 매번 핵심부에 무엇을 포함시키려고 하는가를 물어본다. 이 대면부는 핵심부에 대해서 잘 알고 있거나 이 대면부를 리용하는 구성정보를 처리하는 스크립트들을 가질수 있는 사람들에게는 편리하다. 지령행대면부는 또한 화면에서 제한된 조종을 하기때문에 차림표방식대면부를 리용할수 없는 사람들에게 좋은 기회로 된다. 지령행대면부의 가장 큰 결함은 이전 질문에로 돌아 갈수 없으며 그것을 다시 조사하거나 선택을 변경할수 없는것이다.

문자방식차례표대면부 이 대면부는 따라 가면서 선택항목을 선택하여 조사할수 있고 임의의 령역에로 돌아 갈수 있는 여러가지 준위의 차림표를 리용한다. 이 선택항목은 도형능력을 가진 콤퓨터들에서 리용가능하다. 이 선택항목은 구성하지 않고 직접 호출할수도 있고 X Window대면부를 호출할수 있다.

도형대면부 즉 X방식대면부 이 대면부는 문자방식차림표대면부와 아주 비슷한지만 실제로는 도형대면부이다. 이것은 가장 인기 있는 대면부이지만 X Windows체계를 설치해야 한다.

이 모든 3가지 방법들(지령행식,차림표식, 순수한 도형식)은 프로쎄스들의 콤파일 혹은 재구축과정에 make편의프로그람들이 리용하는것과 같은 구성파일을 작성한다. 또한 지정된 선택항목에 대한 질문에 대답하는 풍부한 방조정보들을 가진다.

선택항목의 리해

임의의 구성방법(지령행식, 차림표식, 도형식)에서 제기된 여러가지 선택항목들을 다시 고찰하려면 매 선택항목이 어떻게 Linux 핵심부에 적용될수 있는가를 리해하여야 한다.

선택항목들은 각이한 지시자로 핵심부를 구성하는데 어느 도구를 리용하겠는가에 따라 지정되지만 같은 선택항목들은 같은 경우에 리용할수 있다. 구성항목들은 2가지 그룹으로 갈라 진다.

- 모듈가능한 항목
- 비모듈가능한 항목

만일 선택항목에 적재할수 있는 핵심부모듈이 없으면 다음과 같은것중의 하나가 되여야 한다.

- [*]는 핵심부의 부분이다.
- [ ]는 핵심부의 부분이 아니다.

우에서 지적한 괄호는 선택항목이 차림표방식구성도구에서 어떻게 나타나는가를 보여 준다.

선택항목이 모줄가능한 항목이라면 이것들을 선택할수 있다(다시말하여 이것들은 차림표구성도구에서 나타나는대로 보여 진다. 즉 비슷한 지시자는 다른 도구들에서도 보여 진다.) 즉

- < >는 후에 적재될수 있는 모듈로서 포함되지도 않고 작성되지도 않는다.
- <*> 는 핵심부의 부분으로 포함되므로 후에 핵심부로 적재되지 말아야 한다. 즉 그것은 항상 체계부분으로 된다.
- <M>는 모듈로 포함되지만 핵심부자체의 부분은 아니다. 그러므로 이 선택항목을 마음대로 추가할수도 있고 실행핵심부로부터 삭제할수도 있다.

일러두기 만일 선택항목을 변경하기 힘들면 먼저 다른 선택항목을 설정해야 한다. 실례로 일반적으로 정의한 SCSI장치지원기능을 선택하기전에 리용할수 있는 SCSI장치지원기능들을 선택할수 있어야 한다.

일단 모든 도구들(make 및 gcc)이 있고 원천코드를 설치했다면 구성도구들의 하나를 실행할수 있고 핵심부를 구성할수 있다.

다음절에서 고찰하게 되는 3가지 대면부가운데서 한개만을 리용해야 한다는것을 명심해야 한다. 매 항목을 간단히 조사하고 주어 진 형태와 해설에 따라 어떤 대면부를 리용하겠는가를 결정해야 한다.

일러두기 처음 새롭게 내리적재된 핵심부를 설치하고 구성하려면 매 항목을 검색해야 한다. 만일 콤퓨터에 대하여 간접적으로 《지정한》것이 있으면(실례로 그것은 하나의 CPU를 가지며 무료형PC Card지원기능, USB지원기능을 요구한다.) 핵심부구성을 변경시켜야 한다.

지령행대면부의 기동

지령행대면부를 기동하려면 다음의 단계들을 거쳐야 한다.

1. 뿌리호출을 위해 su지령을 준다.
2. 원천코드등록부로 변경시킨다. 즉

   ```
   # cd /usr/src/linux-2.4.2
   ```

주의 만일 각이한 등록부에 각이한 핵심부를 내리적재했다면 적당히 바꿀수 있다.

3. 다음과 같은 make지령을 실행시킨다.

   ```
   # make config
   ```

그러면 즉시 첫 구성질문이 나타난다. make config지령을 실행하면 핵심부에 대한 질문들에 응답하도록 재촉문이 나타난다. 그림 20-1은 첫 질문을 보여 준다.

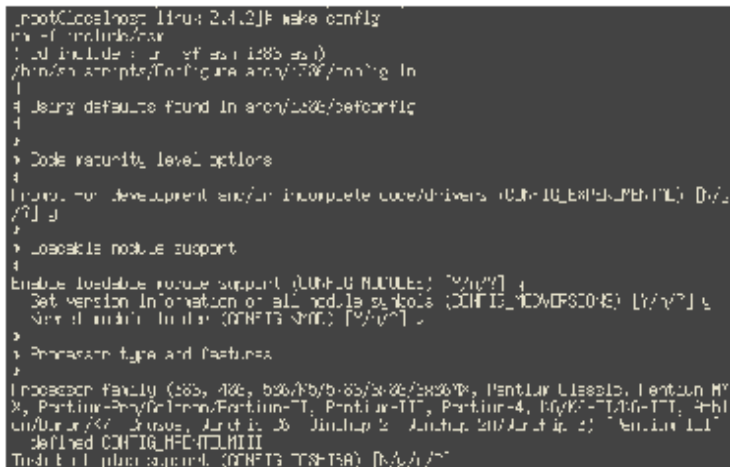



그림 20-1. Linux핵심부에 대한 가장 기초적인 구성항목은 **make config** 이며 지령행형식으로 질문들을 제기한다.

일러두기 마주치게 되는 련속적인 질문들은 질문에 따라 내려 가면서 대답하는 방식에 기초하고 있다.

만일 매 질문에 대하여 기정선택항목을 선정하려면 간단히 Enter건을 눌러야 한다. 매 경우에 기정선택은 대문자로 나타난다. 실례로 망에 대한 질문은 다음과 같다.

Networking support (CONFIG_NET) [Y/n/?]

만일 Enter건을 누르면 Yes에 대한 기정값으로 Y를 선택해야 한다. 조립 혹은 모듈로 정의될수 있는 선택항목들에 대해서는 다음과 같이 나타난다.

kernel support for ELF binaries (CONFIG_BINFMT_ELF)[Y/m/n/?]

장치와 련관된 대부분의 질문은 [Module] 또는 [Don't include]에 대하여 오직 m/n 선택항목만을 제공한다.

매 물음에 대하여 물음표식 ?을 입력하여 해설정보화면을 볼수 있다. 이 정보는 보통 선택항목을 어떻게 리용하겠는가를 결정하는데서 매우 큰 도움을 준다.

일러두기 이와 같은 정보는 3개의 핵심부구성도구들에서 리용할수 있다.

이 구성선택항목의 리용하기는 다음과 같은 3가지 결함들을 가진다.

- 몇백개의 개별적질문들 대답해야 한다(어떻게 대답하는가에 따라 총체적인 수자가 변한다.)
- 기동하지 않고서는 사용자가 선택한것들을 복귀하거나 변화시킬수 없다.
- 여러가지로 분류된 선택항목들에 대한 개괄을 쉽게 볼수 없다.

이 구성선택항목을 리용하면 다음과 같은 유리한 점들이 있다.

- Linux 핵심부원천을 제외하고 핵심부를 구성하는 매우 기초적인 Linux설치만 진행한다.
- 자동구성들을 위한 구성스크립트에 입력을 제공하는 스크립트를 작성할수 있다.

다음 절에서 보게 되는 차림표구성선택항목은 많은 사용자들에게 모든 핵심부구성파라메터들을 조사할수 있고 더 쉽게 핵심부구성파일을 작성할수 있는 한가지 방법을 제공한다.

주의 만일 모든 질문에 대답하기전에 구성처리를 끝내거나 취소하려면 Ctrl+C를 눌러야 한다. 그러면 스크립트로부터 오유통보문이 나타나며 구성파일은 만들어 지지 않는다.

문자방식차림표대면부의 기동

문자방식차림표대면부를 기동하려면 다음과 같은 단계들을 거쳐야 한다.

1. 뿌리호출을 위해 su지령을 리용한다.
2. 원천코드등록부를 다음과 같이 변경시킨다.

```
# cd /usr/src/linux-2.4.2
```

주의 만일 각이한 핵심부를 서로 다른 등록부에 적재했다면 적당히 바꿀수 있다

3. 다음 make지령을 실행시킨다.

```
# make menuconfig
```

주의	차림표방식구성도구는 **ncurses**라고 하는 프로그람묶음을 요구한다. **ncurses**는 많은 Linux체계들에 기정으로 설치되지만 만일 **make menuconfig**지령을 실행시킬 때 오유가 나타나면 **ncurses-devel**프로그람묶음이 설치되여 있는가를 검색해야 한다.

make지령이 차림표방식구성프로그람을 콤파일하는대로 혼돈하기 쉬운 행묶음이 화면에 내리흐르는것을 볼수 있다. 이 행들은 무시할수 있다. 그것들은 콤파일러와 그와 관련된 프로그람들의 출력이다. 1~2분후에 그림 20-2와 같은 화면이 나타난다.

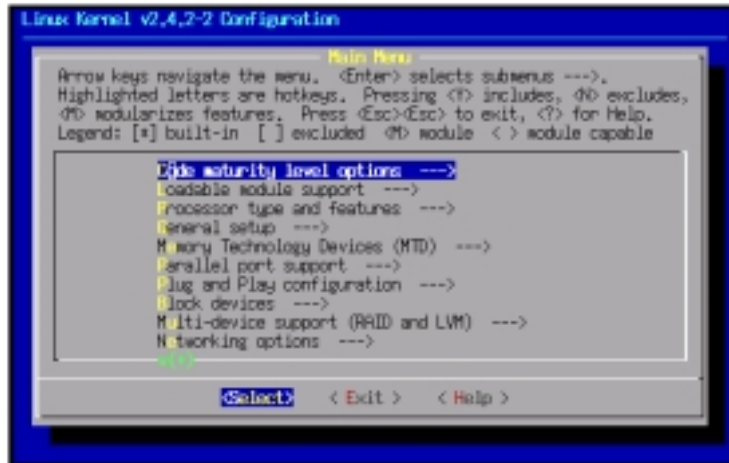



그림 20-2. 이 문자방식차림표대면부로부터 핵심부를 구성할수 있다.

차림표구성선택항목은 핵심부선택항목을 선정할수 있는 분류목록으로부터 시작한다. 이 대면부를 조종하려면 다음의 건조작방법을 알아야 한다.

- 차림표에 제시된 선택항목들은 웃방향 및 아래방향건들을 리용하여 옮길수 있다.
- 모듈로 핵심부선택항목을 선택하려면 M을 입력해야 한다(적용가능할 때).
- Tab건을 리용하여 핵심부선택항목들의 목록아래에 나타난 단추들 즉 Select, Exit, Help사이를 이동할수 있다.
- Help단추와 같이 선정된 항목을 선택하려면 Enter건을 눌러야 한다.

선택항목아래에 있는 3개의 단추들은 다음의 기능들을 가진다.

Select 강조된 항목과 관련된 선택항목들의 보조차림표를 현시한다.

Exit 가장 높은 준위의 차림표에서 선택된 구성을 끝내고 이전 차림표로 돌아 간다.

Help 강조된 차림표선택항목에 대한 정보로 가득찬 도움말화면을 현시한다.
표본적인 도움말화면을 그림 20-3에 보여 주었다.

이 절에서 본 핵심부구성편의프로그람들은 핵심부를 재작성하는데 리용되는 핵심부구성파일을 작성하기 위해서 리용된다. 일단 필요한 모든 선택항목들로 핵심부를 설치했다면 후에 반복하여 리용할수 있게 구성파일의 복사물을 보관해 두는것이 좋다.

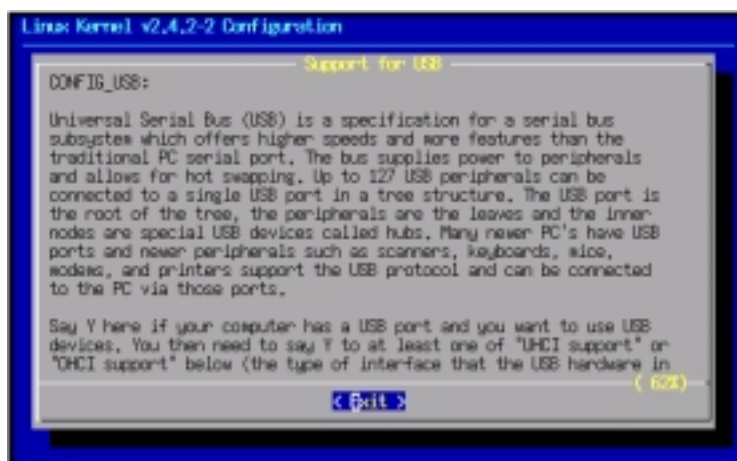



그림 20-3. Help단추를 선택하여 매 핵심부선택항목에 대한 방조정보를 볼수 있다.

이것은 기본차례에 있는 2개 항목 즉 Save Configuration To An Alternate File과 Load An Alternate Configuration File을 리용하여 진행한다. 이 항목들을 내리흐르게 하려면 아래방향건과 Page Down건을 리용해야 한다.

기본차례에 있는 이 2개의 항목들은 구성설정들을 보관하거나 현재 있는 구성파일을 적재할수 있는 능력을 제공한다. 즉

- 다시 리용하거나 다른 사람에게 넘겨 주기 위해 핵심부구성파일의 2번째 복사물을 만들려면 Save Configuration To An Alternate File선택항목을 리용해야 한다.

경고 여러가지 차림표에 있는 모든 핵심부선택항목들의 선택을 완성할 때까지 Save Configuration To An Alternate File선택기능을 리용하지 말아야 한다.

- 만일 누군가가 주었거나 이미 보관한 구성파일이 있다면 Load An Alternate Configuration File선택항목을 리용할수 있다. 앞으로 더 변경시킬것이 없다면 그런 파일을 적재하여 차림표항목중의 임의의것을 선택하지 않도록 모든 선택항목들을 설정해야 한다.

보조차림표를 가지고 있는 차림표목록의 선택항목들은 항목의 오른쪽방향화살표식과 함께 현시된다. 실례로 기본차림표에서 다음과 같은 선택항목을 볼수 있다.

USB Support →

실제로 핵심부선택항목을 설정하는 차림표항목을 보면 이 장에서 이미 본것과 같은 선택항목들이 선택항목의 왼쪽에 현시되는것을 볼수 있다(그림 20-4를 보시오.). 즉

- [*]는 핵심부의 부분이다.
- [ ]는 핵심부의 부분이 아니다.
- < >는 후에 적재될수 있는 모듈로서 포함되지도 않고 작성되지도 않는다.

- <*>는 핵심부의 부분으로 포함되므로 후에 모듈로 적재되지 말아야 한다. 즉 언제나 체계의 부분이다.
- <M>는 모듈로 포함되지만 핵심부 그자체의 부분은 아니다. 따라서 이 선택항목은 실행하고 있는 핵심부로부터 마음대로 추가 혹은 삭제할수 있다. 그림 20-4 에서 강조된 해당한 선택항목 USB Modem은 구성부분으로서 이 모듈을 포함한다. 새롭거나 재콤파일된 핵심부가 설치된후 즉 /sbin/lsmod 지령을 리용한후에 이것을 확인할수 있다. 이와 같은 모듈들은 이미 이 장에서 본것처럼 /sbin/insmod와 /sbin/rmmod지령으로 추가 혹은 삭제할수 있다.

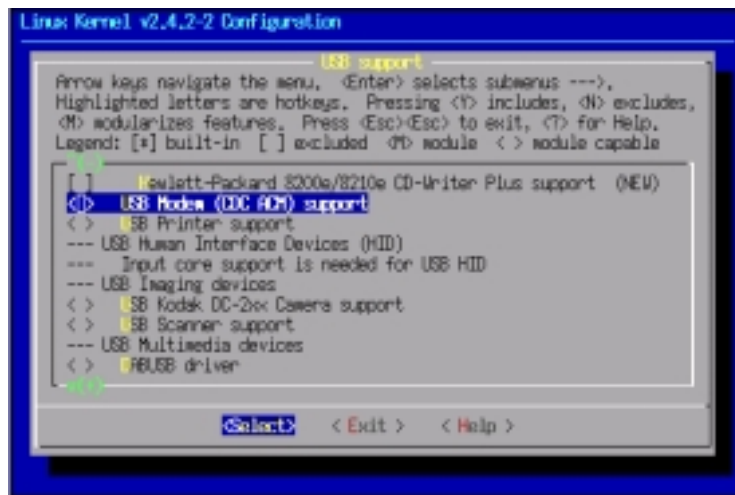



그림 20-4. menuconfig도구에 있는 개별적인 선택항목들은
왼쪽에 놓인 선택괄호와 함께 현시된다.

일러두기 차림표들을 나타내는 화면에 따라 매개 행부분에서 두드러 지거나 색갈이 다른 문자를 볼수 있다. 화살건들과 Select단추를 리용하지 않고 그 문자를 입력하여 선택항목을 선정하는것이 더 좋다.

주의 USB지원에 대한 선택항목들을 구성하는것이 얼마나 쉬운가를 알아야 한다. USB주변장치를 설정하려면 제18장 《주변장치의 리용》에서 고찰한 수법들을 리용하여야 한다.

매개 선택항목의 현재상태는 그 선택항목의 왼쪽에 있는 괄호에 나타난다. 선택된 선택항목을 변경시키려면 다음과 같은 건들을 눌러야 한다.

- Y는 핵심부에 특징을 포함시키기 위한것이다.
- M는 이 특징을 지원(가능할 때)하는 모듈을 작성하기 위한것이다.
- N는 특징을 포함시키지 않거나 그것을 위한 모듈을 만들지 않기 위한것이다.
- ?는 강조된 항목에 대한 도움말화면을 현시하기 위한것이다.

일러두기 강조된 항목에 대한 가능한 선택항목들사이를 고정시키기 위해서 공백건을 반복하여 눌러야 한다.

검색하려는 모든 핵심부선택항목들을 조사하고 설정했으면 기본차림표에 있는 Exit 항목을 선택해야 한다. 그러면 구성화면에는 이 선택들에 기초한 갱신된 핵심부구성파일을 보관하겠는가 하는 질문이 현시된다(그림 20-5). 구성파일을 작성하려면 Yes를 선택해야 한다.

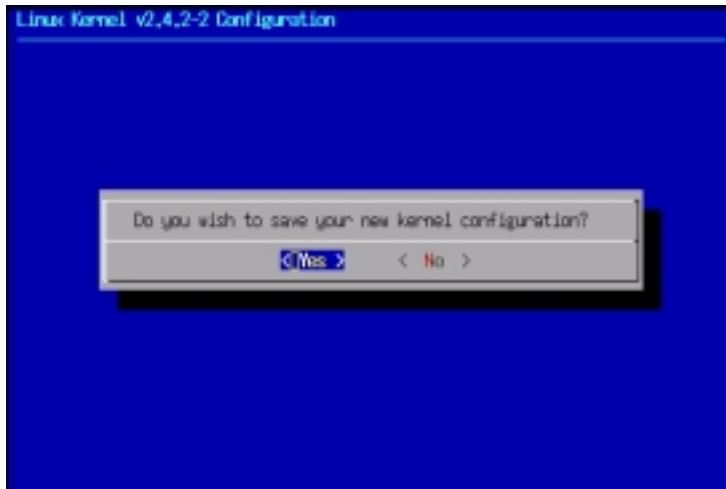



그림 20-5. 구성선택항목들을 선택한후 차레표방식구성도구를 완료할 때 새로운 구성파일을 보관하는데 주의를 돌려야 한다.

도형대면부의 기동

도형(X방식) 핵심부구성대면부를 기동하려면 다음과 같은 단계들을 거쳐야 한다.

1. -X Windows체계를 기동한다(임의의 도형환경 혹은 해당한 탁상을 리용하여).
2. 말단모방기(지령행)창문을 연다.
3. 뿌리호출을 위해 su지령을 리용한다.
4. 원천코드등록부로 변경시킨다.

```
# cd /usr/src/linux-2.4.2
```

주의 만일 각이한 핵심부를 서로 다른 등록부에 내리적재하였다면 적당히 바꿀수 있다.

5. 다음 make지령을 실행한다.

```
# make xconfig
```

주의 X방식구성도구는 몇개의 X Windows체계개발프로그람묶음을 요구한다. 만일 **make xconfig**지령을 실행할 때 오유가 나타나면 알맞는 개발프로그람묶음들이 체계에 설치되였는가를 검색해야 한다.

차림표방식선택항목에서와 같이 make지령이 X방식구성편의프로그람을 콤파일하는대로 화면에 몇개 행이 내리흐른다. 이 행들은 콤파일러와 그와 련관된 프로그람의 출력이다. 1~2분후에 X방식구성프로그람의 기본창문이 그림 20-6과 같이 나타난다.

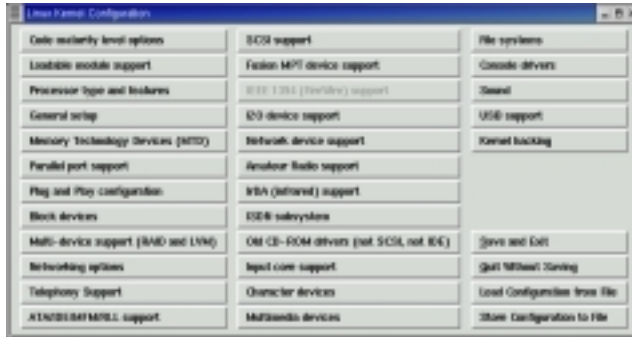



그림 20-6. 도형핵심부구성도구는 선택하는 핵심부선택항목들에 대한 차림표와 단추를 제공한다.

일러두기 도형프로그람은 동등한 문자방식프로그람보다 더 많은 체계자원을 요구한다. 만일 체계의 기억용량이 부족하면 X방식핵심부구성도구는 느리게 동작할수 있다. 그러면 앞의 절에서 본 차림표방식구성도구를 리용해야 한다.

X방식xconfig 도구는 menuconfig 도구의 리용방법과 거의 비슷하다. 선택하려고 하거나 선택하려고 하지 않는 특성들의 부류들을 선정한다. 매 부류에 대해서 핵심부에 특성들이 구축되도록 선택할수 있는 대화칸이 나타나는데 이 특성들은 적재할수 있는 모듈로 작성될수도 있고 전혀 리용되지 않을수도 있다.이 대화칸의 한가지 실례를 그림 20-7에서 보여 주었다.

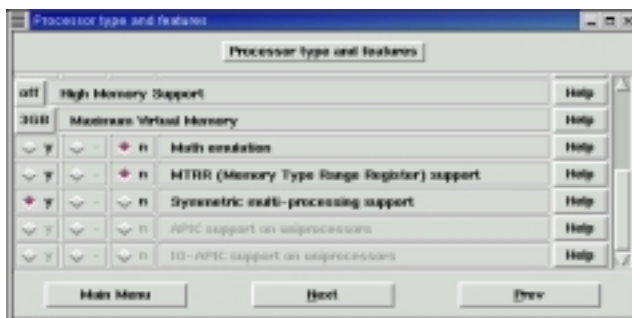



그림 20-7. 이와 비슷한 대화칸은 X방식핵심부구성도구의 특성을 선택하는데 리용된다.

일러두기 Red Hat Linux에 대한 핵심부 2.4.2를 갱신하면 균형적인 다중처리지원기능을 기정으로 쓸수 있다. 실제로 콤퓨터에 CPU밖에 없으면 이 설정을 n으로 변경해야 한다.

여러 특성들을 다시 조사하면서 그 특성들에 대한 도움말화면을 보기 위해 항목의 오른쪽에 있는 Help단추를 선택할수 있다(그림 20-8). 이 화면에 제공된 정보는 지령행식 혹은 차림표방식구성도구들에서 볼수 있는것과 같다.

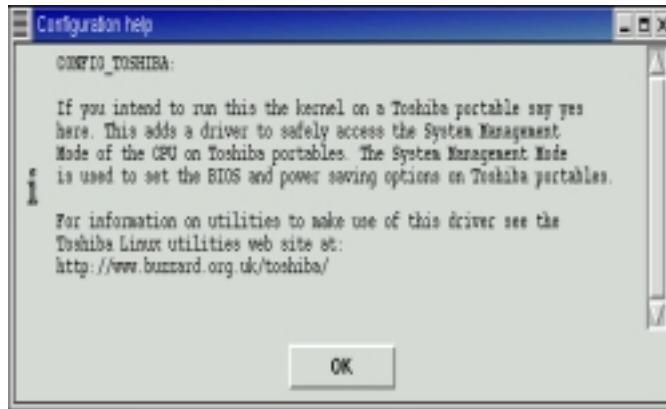



그림 20-8. X방식구성도구를 리용할 때 방조정보는 개별적인 창문에 나타난다.

새로운 핵심부구성으로 처음 작업할 때 X방식구성도구를 리용하면 도형대면부가 여러가지 선택항목들사이의 종속기능들을 보여 주는데 이것이 X방식구성도구리용의 가장 큰 우점이다. 실례로 Block Device항목에서 RAID지원선택항목들이 핵심부에 구축되기전에 Multiple Device Driver Support 선택항목을 리용할수 있어야 한다.

menuconfig도구는 차림표에 있는 같은 글을 리용하여 종속성을 보여 주지만 도형도구는 RAID에 따르는 Multiple Device Driver Support선택항목을 선택할 때까지 RAID 선택항목을 무시한다. 결과 X방식구성도구를 리용하여 모듈과 특성사이 관계를 더 쉽게 리해할수 있다.

X방식도구아래에 있는 선태항목들은 차림표방식기초도구에서 리용한것과 비슷하다. 즉

Save And Exit 핵심부구성파일을 작성하고 구성도구를 닫는다.

Quit Without Saving 구성도구를 닫지만 핵심부구성파일을 작성하지 않는다.

경고 만일 이 장에서 서술한 3가지 도구중 하나를 리용하여 핵심부구성도구를 작성하지 않으면 핵심부를 재조립할수 없다.

Load Configuration From File 선택한 이름으로 기억된 이미 보관한 구성파일을 적재한다(후에 리용하거나 친구에게 보내기 위해서).

Store Configuration To File 핵심부구성파일을 선택한 이름으로 보관한다(후에 리용하거나 혹은 친구에게 보내기 위해서). Save And Exit 선택을 리용하여 기정위치에서 즉시 핵심부를 다시 구축하는데 리용될수 있는 핵심부구성파일을 작성해야 한다.

새로운 핵심부의 콤파일과 실행

일단 핵심부재구성이 완성되면 새 구성파일(.config)이 /usr/src/linux-2.4.2등록부에 있는가를 검색해야 한다. 숨은 파일을 보려면 ls -a지령을 리용해야 한다. 만일 .config가 있으면 make지령으로 핵심부를 재구성할 준비를 한다.

앞서 본 지령들은 Linux체계에 따라 약간 다를수 있지만 거의 비슷하다.

핵심부를 구성한후 지령행으로 돌아 가서 콤파일하기 위해 체계가 제대로 기동하고 있는가를 확인해야 한다.

핵심부재조립은 처리소자, 기억기, 기타요소에 따라 15분으로부터 몇시간 걸릴수 있다. 그렇기때문에 대다수의 사용자들은 핵심부를 재조립하는 모든 지령들을 한개의 지령행에 반두점으로 구분하여 결합해 놓음으로써 지령들을 차례로 실행시킨다. 이렇게 하면 잠시후에 체계에로 돌아 갈수 있으며 전 과정이 끝났음을 알수 있다.

조립과정의 시작

여기서 주어 진 지령들은 새 핵심부를 작성하고 표준모듈지령들을 리용하여 핵심부가 호출할수 있는 정확한 체계등록부들에 핵심부모듈들을 재조립하여 놓는다.

체계를 재조립하려면 다음과 같은 지령들을 리용하여야 한다.

```
# make dep; make clean; make bzImage; make modules; make
modules_install
```

일러두기 매 make지령들은 하나씩 개별적으로 입력할수 있다. 만일 모듈들을 리용하지 않겠다면 마지막 두개의 지령을 뺄수 있다. 더 많은 정보를 보려면 새 핵심부를 풀어 놓은 곳(/usr/src/linux-2.4.2)에서 README파일을 조사하여야 한다.

이 지령들을 입구하면 make프로그람이 각이한 등록부들에 들어 가서 각이한 원천코드파일을 콤파일하는 gcc나 egcs를 실행하고 그 코드쪼각들을 하나로 결합하는 과정의 정보들이 화면에 내리전개되는것을 볼수 있다. 이 매 지령들을 완성하는데 몇분이 걸린다.

만일 /etc/lilo.conf파일에 initrd파일이 있다면 이 경우에 새 initrd파일을 작성해야 한다. 먼저 현재 initrd파일을 여벌복사하는데 그 파일은 Red Hat Linux 7.1의 initrd-2.4.2-2.img이다. 이렇게 하려면 다음의 지령들을 실행시켜야 한다.

```
# cp /boot/initrd-2.4.2-2.img /boot/initrd-2.4.2-2.orig.img
# /sbin/mkinitrd /boot/2.4.2-2.img 2.4.2-2
```

이제는 다음의 지령으로 새로운 핵심부로부터 기동디스크를 만들수 있다.

```
# make bzdisk
```

이 지령을 실행하기전에 구동기에 초기화된 플로피디스크를 넣어야 한다. 일단 기동디스크가 준비되였으면 그것을 검사해야 한다. 그러자면 플로피구동기의 기동디스크로 콤퓨터를 재기동해야 한다.

일단 이미 서술된 지령들이 완성되여 지령행으로 돌아 가면 체계를 기동할수 있는 새로운 핵심부를 가지게 된다.

새롭게 작성된 핵심부를 표준위치에 옮겨서 리용할수 있게 해야 한다. 이것을 위한 지령은 다음과 같다.

```
# cp /usr/src/linux-2.4.2/arch/i386/boot/bzImag /boot/vmlinuz-2.4.2-2
```

마지막으로 기동적재작성표를 갱신하기 위하여 lilo지령을 실행시켜야 한다. 즉

```
# /sbin/lilo
```

일러두기 만일 복사된 핵심부영상이름이 다른 핵심부판본번호를 가진다면(혹은 짝수아닌 번호) 알맞게 바꿀수 있다. 앞의 절에서 핵심부의 여벌복사에 대해서 본것처럼 /etc/lilo.conf파일에 주어 진 해당한 파일이름과 맞는가를 비교해 보아야 한다.

새로운 핵심부의 검사

새로운 핵심부를 기정위치(lilo.conf파일에서 보여 준것처럼)로 옮긴후 새로운 핵심부를 기동시키기 위하여 체계를 재기동할수 있다. 일단 Linux체계가 재기동되였으면 핵심부구성에서 정의한 추가적인 특성들을 시험해 보아야 한다. 여기에는 다음과 같은것들이 포함된다.

- 새 핵심부와 이전 핵심부의 크기를 비교한다. 또한 체계기억기가 얼마만큼 리용되였는가를 보기 위해 free지령을 실행한다.
- 처음에 장치를 지원하는 핵심부모듈을 적재하지 않고도 파일체계를 설치하거나 장치를 호출할수 있다(이러한 봉사지원기능이 있다면).
- 일반적인 핵심부에서 리용할수 없었던 망자원(IP주소와 같은)을 리용한다.

uname지령으로 현재 핵심부의 작성시간을 볼수 있다. 이 지령은 실행하고 있는 핵심부가 재조립한 핵심부인가를 확인한다. 핵심부를 재조립한 다음 날자와 시간이 일치하는가를 확인해야 한다. 즉

```
# uname -v
#1 Tue Mar 9 13:27:39 EST 2001
```

주의 만일 **uname**지령으로부터 응답이 새 핵심부가 시동하고 있지 않다는것을 제시하면 LILO시동적재기에 문제가 있다. /etc/lilo.conf파일에 있는 정보를 조사하여 핵심부가 정확히 정의되였는가를 확인해야 한다.

▶미리 알아두기

다음부분부터는 기본접속을 고찰한다. 다음장들에서는 PPP접속을 구축하고 Linux로 World Wide Web를 열람하며 Linux로 전자우편을 주고받는 기구들을 비롯하여 개별적인 PC와 인터네트를 접속시키는것과 관련된 분야들을 고찰한다.

제21장 《Linux망련결에 대한 리해》는 Linux에서 활용하는 기초적인 망개념에 대한 고찰로부터 시작한다. 여기서는 인터네트망을 작성하고 대부분의 Unix와 Linux용망들에 대한 기초들인 규약 TCP/IP을 고찰한다.

제5편. 기본접속

제21장. Linux망련결에 대한 리해

이 편에서는 매일 망작업을 진행하는 많은 콤퓨터사용자들에게 흥미를 주는 화제를 고찰한다.

이 장은 TCP/IP망들과 경로작업을 비롯한 기본Linux망작업개념을 고찰하는것으로부터 시작한다. IP주소, 망마스크, 포구, 관문과 같은 TCP/IP구성의 구성요소들과 TCP/IP의 기본개념들을 고찰한다. 그다음에 표준Linux체계의 TCP/IP결합으로 전문실행하는 봉사형태들을 고찰한다.

TCP/IP기초

TCP/IP는 인터네트망에 대한 콤퓨터통신언어이다. 인터네트의 력사를 보면 TCP/IP는 Unix에서 개발되였다. Linux가 Unix로부터 파생되였기때문에 Linux는 인터네트에 접속할수 있는 하나의 큰 조작체계이다.

TCP/IP와 같은 언어는 콤퓨터호상간 통신에 리용되는 규약들의 묶음이다. 이 규약들이 망에 있는 다른 체계들과 물리적인 매체로 통신하기때문에 구축된 망에 놓이는 독자적인 형태의 물리적인 결합이다.

물리적인 준위에서 망들은 전화회선이나 TV동축케블, 빛섬유케블 등을 비롯하여 다양한 수단들을 리용하면서 구축된다. 사용자는 다양한 물리적인 매체상에서 TCP/IP 혹은 기타 망규약묶음들을 실행할수 있다.

TCP/IP란 무엇인가?

TCP/IP는 많은 규약들에서 중추로 되는 2개의 이름으로 구성되였다. 이 규약들은 오늘 업무용으로 나온 거의 모든 조작체계들과의 통신에 리용되고 있다. TCP/IP는 호상작용성을 가지기때문에 망에서 리용할수도 있고 인터네트에 접속할수도 있다.

TCP/IP구성의 본질

TCP/IP가 어떻게 구성되고 망들이 어떻게 설계되는가를 알려면 몇가지 기초적인 개념을 리해해야 한다. 즉

- IP주소
- 부분망과 망마스크
- 방송망(Broadcast)주소
- 관문주소
- 이름봉사기

이 정보는 IP판본 4(IPv4)에 기초하고 있다. 현재 IP판본 6(IPv6)으로 이전하고 있지만 새로운 판본의 IP주소화방식은 IPv4의 대부분의 중요한 개념들을 포괄하고 있다.

IP주소

TCP/IP의 세계에서는 매개 콤퓨터를 주콤퓨터라고 한다. 매개 주콤퓨터들은 TCP/IP망에 접속될 때 IP주소라고 하는 유일한 주소를 가지게 된다.

한개의 IP(IPv4)주소는 점으로 구분된 0～255범위에 있는 4개의 수를 가진다. 인터네트에 직접 접속된 매개 콤퓨터들은 유일한 IP주소를 가진다. 인터네트봉사제공자(ISP)를 거쳐 인터네트에 전화망으로 접속할 때 ISP는 그 콤퓨터에 유일한 IP주소를 할당한다.

콤퓨터들은 1과 0의 2진코드로 통신한다. 192.168.0.34같은 IP주소를 2진코드로 변환하면 다음과 같다.

11000000 10101000 00000000 00100010

이 32개의 매개 수자를 2진수 또는 비트라고도 한다. 보는바와 같이 IP주소는 32bit로 되여 있다. 32bit의 수자에서 1과 0의 가능한 조합의 총수는 2^{32} 즉 4,294,967,296이다. 얼핏 보기에 이것은 대단히 많아 보이지만 실제로 인터네트에서는 IP주소가 고갈되여 가고 있다.

후에 보게 되는 IP판본 6(IPv6)은 이 문제를 해결하기 위하여 개발되였다. IPv6에서 IP주소들은 128bit들로 되여 있다. 128bit주소에서 1과 0의 가능한 조합의 총수는 2^{128} 즉 340, 282, 366, 920, 938, 463, 374, 607, 431, 768, 211, 456이다.

대충 짐작컨대 40억개이상의 주소들은 세계적규모에서 순전히 우연적으로 리용되지 않는다. 오히려 그 주소들은 인터네트와의 리용에서 IP주소들을 요구하는 조직, ISP, 다른 그룹이 리용하는 망이라고 하는 묶음으로 할당된다.

정규적인 망접속에서 리용할수 있는 TCP/IP망들에는 3가지 형태가 있다. 즉 A급망, B급망 및 C급망이 있다.

- A급에서 망은 IP주소의 첫 바이트로 구별된다. 나머지 3개 바이트는 망에서 지정한 콤퓨터들을 구별한다. 총 16,777,214개의 주소들이 가능하다. A급에서 첫번째 수자는 1～126범위에 있을수 있다. 실례로 A급에서 98로 시작하는 망의 가능한 IP주소는 98.0.0.1～ 98.255.255.254범위에 있을수 있다.
- B급에서 망은 IP주소의 첫 2개 바이트로 구별된다. 나머지 2개 바이트는 망에서 지정한 콤퓨터를 구별한다. 총 65,534개의 주소들이 가능하다. B급망에서 첫 수는 128～191사이에 있다. 두번째 수는 0～255사이의 값을 가진다. 실례로 145.255로 시작하는 B급망을 보면 이 망에서 가능한 IP주소들은 145.255.0.1 ～145.255.255.254사이에 놓일수 있다.
- C급에서 망은 IP주소의 첫 3개 바이트로 구별된다. 나머지바이트는 망에서 지정한 콤퓨터를 구별한다. 총 254개의 주소들이 가능하다. C급에서 첫 수자는 192～223사이의 값을 가진다. 2번째와 3번째수는 각각 0～255사이의 값을 가질수 있다. 실례로 212.230.0으로 시작하는 C급망을 보면 가능한 IP주소들은 212.230.0.1～ 212.230.0.254사이에 놓일수 있다.

우의 범위에서 첫 수자와 마지막수자(0 및 255)는 제외되였다.이것은 령역에서 첫 수가 망주소로 예약되여 있고 마지막수가 방송망주소로 예약되여 있기때문이다. 실례로 주목된 A급망주소는 98.0.0.0이고 방송망주소는 98.255.255.255이다.

또한 IP주소에서 0, 127, 224～255를 비롯하여 몇개의 수는 언급하지 않았다. 비공개리용을 위하여 예약된 몇가지 주소들도 있다. 보다 많은 정보들에 대해서는 다음절의 주의들에서 제시된 문서들을 참고할수 있다.

부분망과 망마스크

큰 IP주소를 요구하는 조직들이 B급망을 선택하는것을 흔히 볼수 있다. 이 조직들은 그것을 256개의 C급망으로 재분할한다. 이에 대한 완전한 실례로 협동의뢰기들에 C급망을 제공할것을 요구하는 큰 ISP를 들수 있다. 이것은 B급망을 분할함으로써 수행할수 있다.

실례로 ISP가 165.65로 된 B급망을 가진다면 ISP는 C급망을 165.65.0~ 165.65.255까지의 범위에 놓이는 256개의 각이한 망으로 나눌수 있다. 이 매개 부분망들은 254개의 리용가능한 주소를 가진다. 이것은 마감절에서 보게 되는 C급망의 설명과 비슷하다.

이 상황에서는 혼란이 일어 날 가능성이 크다. 어느 망에 있는가를 콤퓨터가 어떻게 알수 있는가? 콤퓨터가 IP주소로 19.148.43.194를 가진다면 그 주소가 A급망의 19인지 아니면 B급망의 19.148인지 아니면 C급망의 19.148.43인지 인차 보고 알수 없다.

이러한 문제는 망마스크(혹은 부분망마스크)를 리용하여 해결할수 있다. 망마스크는 점으로 분리되고 IP주소의 어느 부분이 망을 구별하는가를 정의하는 한 바이트옹근수묶음이다.

일반적으로 망마스크에는 세개의 값 즉 255.0.0.0, 255.255.0.0, 255.255.255.0이 있다. 만일 IP주소와 망마스크를 알고 있는 상태에서는 망IP주소와 리용가능한 주소범위를 정의할수 있다.

앞에서 본 IP주소 19.148.43.194를 취한 상태에서 망마스크가 255.0.0.0이라면 망주소는 19.0.0.0으로 되며 그 망에서 가능한 주소의 범위는 19.0.0.1~ 19.255.255.254사이에 놓인다. 망마스크가 255.255.0.0이라면 망주소는 19.148.0.0으로 되며 그 망우에서 가능한 주소의 범위는 19.148.0.1~ 19.148.255.254까지로 된다. 망마스크가 255.255.255.0이라면 망주소는 19.148.43.0으로 되며 그 망우에서 가능한 주소의 범위는 19.148.43.1~ 19.148.43.254이다.

어떤 콤퓨터가 통보문을 경로조종하면 그것은 특수한 IP주소를 가진다. 후에 보게 되겠지만 그 콤퓨터에 망마스크가 있으면 그 IP주소를 어디에 보내야 하는지를 알수 있다.

주의 더 복잡한 망마스크들을 리용할수 있지만 이 책에서는 언급하지 않는다. 주소블로크를 나누는데는(혹은 재결합) 많은 방법이 있다. 이 문제에 대한 더 구체적인 정보는 http://www.linuxdoc.org/HOWTO/mini/IP-Subnetworking.html의 IP Sub-networking mini HOWTO를 참고하거나 http://www.linuxdoc.org/LDP/nag2/index.html의 *Linux Network Administrators Guide*에 대한 제2장을 참고할수 있다. 자기의 TCP/IP망을 계획하고 있다면 개인주소에 대한 설명에 주목을 돌려야 한다.

IP판본 6(IPv6)

최근에 2^{128}의 주소를 가진 IPv6으로 이전하고 있다. 이러한 전환은 단번에 진행되지 않으며 망하드웨어와 쏘프트웨어에서 IPv4와 IPv6주소를 동시에 둘다 리용할수 있도록 개발하고 있다. IPv6주소의 자리수가 많기때문에 모든 IPv4주소들을 포함할수 있는 충분한 방이 있다. 현재 리용중에 있는 IP주소(IPv4)는 IPv6에서도 그대로 리용할수 있다.

사실상 IPv6을 위해 개발된 규약들은 조금 차이를 가진다. 실례로 IPv4주소

192.168.33.54

는 다음의 IPv6주소와 같다.

::192.168.33.54

그 변환이 완성되면 16진수 즉 16기준표기법으로 IPv6주소를 생각하게 된다. 16기준표기법에서는 수값으로 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f를 리용한다.

16진수표기법으로 전형적인 IPv6주소를 다음과 같이 표시할수 있다.

3dfe:0b80:0a18:1def:0000:0000:0000:0287

IPv6규약의 다른 표기방법은 령을 떼버리는것이다. 우와 같은 주소를 다음과 같이 표시할수도 있다.

3dfe:b80:a18:1def:0:0:0:287

앞에서 표기했던 IPv4주소를 바꿀수도 있다. 이를 위해 먼저 그것을 2진수로 변환한다. 즉 192.168.33.54는

11000000 10101000 00100001 00111000

으로 된다. 이것을 16진수표기법으로 변환하면

c0a8:2138

로 된다. 그러므로 완전히 대응하는 IPv6주소는

0000:0000:0000:0000:0000:0000:c0a8:2136

로 되거나

0:0:0:0:0:0:c0a8:2136

또는

::192.168.33.54

로 된다.

방송망주소

방송망주소는 망에서 모든 주콤퓨터들에 정보를 전송할 때 리용할수 있는 특수한 주소이다. 즉 매 주콤퓨터에 개개의 통보문으로 보내지 않고 단일통보문을 방송망주소에 보낼수 있다. 그 망에 련결된 모든 콤퓨터들은 이 통보문을 수신한다. 라지오나 텔레비죤신호가 방송망통보문에 가장 류사하며 청취하는 모든 단위들에서는 이 통보문을 수신한다.

방송망주소는 망주소와 함께 255가 있는 주콤퓨터부분으로 이루어 진다. B급망 174.148에 대한 방송망주소는 174.148.255.255로 되며 C급망 194.148.43에 대한 방송망주소는 194.148.43.255로 된다.

관문주소

경로조종에 대한 절에서 보게 되겠지만 국부망 혹은 부분망을 구성하는 콤퓨터는 외부망(실례로 인터네트)에 있는 콤퓨터들과 통신하는 방법을 알지 못한다. 이를 위해 필요한것이 바로 관문이다.

관문는 외부세계에 대한 경로를 제공하는 콤퓨터이다. 일반적으로 관문는 적어도 국부망접속과 외부망접속의 두가지 망방식을 가진다. 국부망에서 콤퓨터가 어떤 통보문의 목적지주소를 알지 못하는 경우 그 통보문을 관문에 보낸다. 관문는 그 통보문을 외부망에로 다시 경로조종한다.

어떤 주콤퓨터가 외부망과 접속하자면 인터네트와 같은 외부망과 접속한 적어도 한개의 관문의 IP주소를 알고 있어야 한다.

이름봉사기

Sybex Web싸이트를 호출하려면 자기열람기의 주소란에 63.86.158.42를 입구해야 한다. 그러나 http://www.Sybex.com을 기억하는것이 더 헐하다. 이름봉사기는 그 변환을 제공한다. 도메인이름체계(DNS)내에서 이름봉사기는 www.mommabears.com 혹은 www.Sybex.com과 그에 대응하는 IP주소와 같은 도메인이름들의 자료기지이다.

인터네트나 어떤 다른 큰 규모의 망과 접속한 콤퓨터에는 이름봉사기가 필요하다. 그러므로 어떤 큰 망우에 있는 주콤퓨터에서 TCP/IP를 구성할 때 이름봉사기를 요구한다.

ISP들은 인터네트에 접속하기 위한 이름봉사기들을 제공한다. 사실상 전화접속방식에서는 이름봉사기의 IP주소를 알 필요가 없다. 그것은 접속시에 자동적으로 제공된다. 전화접속방식에 대해서는 제22장 《Linux를 인터네트에 접속》에서 더 구체적으로 설명한다.

TCP/IP포구

두대의 콤퓨터사이에 TCP/IP를 리용하여 통신하려면 일반적으로 IP주소와 포구번호가 필요하다. 실례로 World Wide Web은 포구 80을 리용한다. http://www.mommabears.com을 입구하면 TCP/IP는 WWW을 찾아서 자동적으로 이것을 http://www.mommabears.com:80로 변환한다.

65,536(=2^{16})개의 리용가능한 포구가 있다. 첫 1,024포구와 다른 포구들은 www, FTP 등과 같은 TCP/IP봉사들에 할당된다. 널리 쓰이는 봉사들인 TCP/IP, Web, FTP, 전자우편에는 모두 포구가 할당된다. 표 21-1에서는 몇가지 일반적인 TCP/IP포구를 보여 준다.

표21-1. 일반 TCP/IP포구번호

봉사이름	포구	형	설 명
FTP	21	TCP	파일전송규약
Telnet	23	TCP	Telnet접속
SMTP	25	TCP	간단한 우편전송규약
Name	42	TCP	도메인이름체계규약
HTTP	80	TCP	하이퍼본문전송규약(World Wide Web)
POP3	110	TCP	우편사무규약3 우편읽기장치
IMAP	143	TCP	인터네트통보문호출규약 우편읽기장치

TCP와 UDP포구의 완전한 목록은 http://www.isi.edu/in-notes/iana/

assignments/port-numbers에서 찾을수 있다. Web봉사기들은 포구 80을 대기한다. http://www.mommabears.com의 Momma Bears'Bears에 들어 가려면 관련된 Web사용자봉사기는 통보문을 받고 Momma Bears'Web싸이트에 통보문을 반환해야 한다.

그러나 Momma Bears'Web봉사기는 포구80에서 대답하지 않는다. 만일 그렇게 되면 Momma Bears'Web싸이트에 다른것들이 도달하는것을 방지하면서 그 통로(포구 80)를 막을수 있다. 다행히도 http://www.mommabears.com을 입구하면 TCP/IP는 Momma Bears'Web싸이트에 할당되지 않은 포구번호를 보낸다. 봉사기는 Web싸이트를 주콤퓨터에로 보내기 위하여 이 포구번호를 리용한다.

경로조종개념

앞에서 관문를 론의하면서 경로조종의 개념에 대하여 언급하였다. 인터네트에서 경로조종은 우편번호와 같다. 즉 통보문을 통하여 자기의 목적지를 찾게 한다.

결국 Yahoo의 Web싸이트에 접속하기 위하여 Netscape나 Web열람기를 리용하면 Yahoo의 실제적인 물리위치에 대하여 알수 있다. ISP는 www.yahoo.com을 IP주소로 변환한다. 어쨌든 Yahoo는 그것을 요구하고 사용자리면에서 자기의 통로를 찾는다.

경로조종의 간단한 실례를 보기로 하자. 콤퓨터가 인터네트와 접속한 어떤 작은 회사의 국부망(LAN)과 접속하였다고 하자. 인터네트와의 접속은 회사 LAN과 접속한 경로기와 자기의 ISP를 통하여 이루어 진다. 매 경로기는 관문으로 동작한다. 관문과 마찬가지로 경로기는 국부망에 대한 접속주소와 IP주소를 가진다. 또한 ISP에 의해 할당된 IP주소를 가진다. 그림 21-1에 이 망의 구성을 보여 주었다.

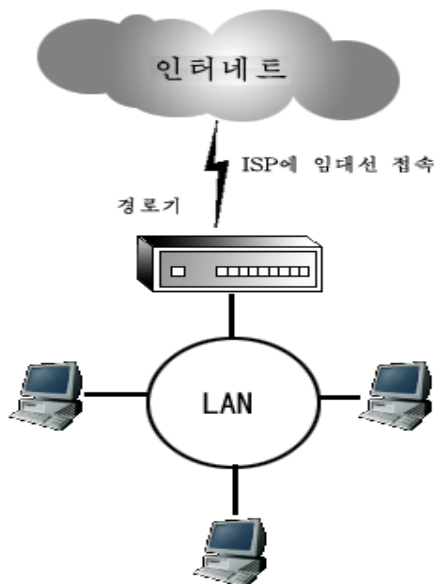



그림 21-1. 인터네트에 접속된 LAN

LAN이 경로기를 통해서만 외부세계와 접속할수 있는것으로 하여 경로조종은 간단한 문제이다. 인터네트에서 매개 콤퓨터는 LAN상의 경로기IP주소를 가진 기정관문으로

구성된다. 언제든지 LAN상의 주콤퓨터가 국부망밖의 콤퓨터와 접속하려면 관문의 IP주소에 대하여 질문한다. 즉 경로기에로 전송한다. 경로기는 ISP에로 정보를 전송한다.

이러한 방법으로 전체 외부세계는 검은 통으로 인식되며 외부를 향한 모든 정보는 마치 경로기가 LAN을 벗어 난 전체 망을 밀봉한것처럼 보고 간단히 경로기에로 전송한다.

더 나아가서 그림21-2에서 보여 준것처럼 ISP의 망과 그의 인터네트접속을 포함하도록 검은 통을 확장해보자.

좀 더 흥미 있는 문제인 ISP의 LAN을 고찰해 보자. ISP의 LAN에는 두개의 경로기가 배치되여 있는데 경로조종기A는 ISP의 LAN을 국부망에로 접속하기 위한것이며 경로기B는 ISP의 LAN을 인터네트에로 접속하기 위한것이다.

매개 망은 기정관문를 요구하므로 경로기B는 ISP의 기정관문이다. ISP의 LAN상에서 주콤퓨터가 인터네트에 접속하려면 경로기B에 정보를 전송해야 한다.

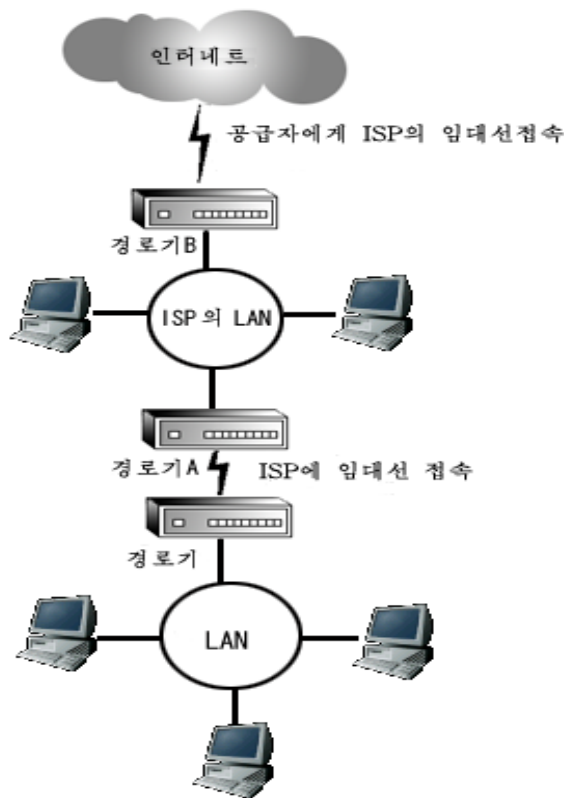



그림 21-2. LAN과 그의 ISP

다시말하여 LAN상에 있는 콤퓨터로부터 오는 통보문은 먼저 사용자측의 기정관문 즉 경로기에 전송된다. 이 경로기는 사용자의 ISP에 통보문을 전송한다. 사용자의 ISP는 경로기A로 그 통보문을 수신한다. ISP는 통보문의 목적지주소를 검사한다. 만일 목적지주소가 인터네트상에 있는것이라면 ISP는 자기의 기정관문인 경로기B에 그 통보문을 전송한다.

경로조종은 다중싸이트, 다중LAN, 인터네트에 대한 다중접속을 가진 거대한 조직들에서 아주 복잡한 양상을 띨수 있다. 이러한 경우에 경로조종은 가능한껏 가장 안전한 방법으로 정보의 가장 효률적인 경로조종을 담보할수 있도록 설계되여야 한다. 실례로 서로 멀리 떨어 져 있는 두 지점을 인터네트로 접속하지 않고 직접 접속시킬수도 있다. 이것은 공중인터네트를 통한 방법으로서 이 싸이트들사이의 프로그람묶음에 대한 경로조종을 느끼지 못하게 한다. 이 방법은 비효률적이고 불필요한 안전상 결함이 있다.

거대한 망들에서의 복잡한 경로조종에 대해서는 이 책에서 설명하지 않으며 망에 대하여 구체적으로 서술한 책을 참고해야 한다. 이 장에서 본것은 대부분의 Linux사용자들이 자체로 고찰할수 있는 전형적인 망경로조종환경을 리해하는데 충분한 배경을 제공하는것들이다.

▶미리 알아두기

이 장에서는 TCP/IP망과 경로조종에 대한 기초개념을 보았다.

제22장 《Linux를 인터네트에 접속》에서는 앞에서 언급한 이러한 리론적인 지식을 실천으로 바꾸어 나간다. 인터네트봉사제공업자들이 제공하는 전화접속방식 PPP접속을 통해서 Linux PC를 어떻게 인터네트에 접속하는가를 보게 된다.

다음장들에서는 Linux로부터 World Wide Web나 전자우편과 같은 특수한 인터네트봉사들을 어떻게 리용하는가를 고찰한다.

제22장. Linux를 인터네트에 접속

일반적으로 Linux체계(혹은 임의의 콤퓨터체계)를 인터네트에 접속하는방법은 좀 복잡하기때문에 망체계환경에 대한 깊은 지식이 요구된다.

그러나 처음으로 X Windows와 Linux체계를 리용하는 많은 사용자들은 아주 간단하게 작업할수 있다.

이 장을 학습하면 인터네트세계에서 PPP와 그의 역할에 대한 지식을 빨리 습득할수 있다. 여기서는 인터네트에 접속하기 위한 하드웨어와 쏘프트웨어의 요구조건을 다시 고찰하며 수동으로 PPP와 접속할수 있는 도구들을 본다. 마지막으로 이 장에서는 접속을 자동화하는 방법을 고찰한다.

PPP란 무엇인가

대다수의 인터네트사용자들은 PPP라는 단어를 잘 알고 있다. 그것은 바로 사용자들이 인터네트봉사제공자(ISP)와 함께 가진 회계의 형태가 PPP회계이기때문이다. 그러나 실제로 대다수의 사용자들은 PPP의 구체적인 기능과 역할에 대해서는 잘 모른다.

PPP는 지점간 통신규약(Point-to-Point Protocol)을 의미하며 TCP/IP가 모뎀을 통하여 확장되는 경우 구체적인 방법들을 규정하고 있다. 이 방법으로 PPP를 리용하여 인터네트에 접속하려면 그 사용자는 ISP의 부분망에 소속되여야 하며 인터네트에서 능동주콤퓨터로서 자기의 IP주소를 가져야 한다.

전통적인 전화접속방식의 인터네트접속은 대체로 말단쏘프트웨어를 리용하며 Unix 쉘은 중심봉사기들을 회계한다. 이러한 환경에서 말단들은 순전히 봉사기에 대한 현시장

치로서만 동작하였으며 그 봉사기만이 인터네트에서 IP주소를 가졌다. 이것은 모뎀을 통하여 인터네트와 접속하는 오늘날의 PPP접속과는 완전히 대조되였다.

이러한 풍부한 인터네트접속기술은 PPP접속형태들이 매우 다양하다는것을 말해 준다. PPP접속에는 고정적인 IP주소와 동적인 IP주소가 있다. 이와 같은 접속기술은 PAP(Password Authentication Protocal) 혹 은 CHAP(Challenge Handshake Authentication Protocol)와 같은 특수한 인증규약들을 리용하고 있다. 지어 그것들은 서로 질문하고 응답하는 맞잡이기능에 기초한 본문형식의 표준통보문들도 리용할수 있다. 수동으로 접속할수 있으며 필요하다면 자동적으로도 할수 있다.

이 장에서는 가장 일반적인 인터네트접속방법들을 고찰한다. 즉 동적IP주소를 가진 모뎀을 PPP접속을 지원하는 인터네트봉사제공자(ISP)에 어떻게 접속하는가 하는 방법들을 고찰한다.

주의	《PPP》와 《ISP접속》은 이 장에서 서로 호환할수 있다. 그러나 PPP접속은 ISP를 통한 접속에는 제한되지 않는다. 사실상 PPP는 회사나 교육망에 접속하기 위한 일반적인 방법이기도 하다.

하드웨어와 쏘프트웨어에 대한 요구

순전히 PPP로 작업하기 위해서는 몇가지를 준비해야 한다. PPP로 작업하기전에 하드웨어와 쏘프트웨어를 알맞게 구성해야 한다. 그러자면 세가지 기본구성요소들이 고려되여야 한다.

- 모뎀이 설치되고 구성된 상태에서 작업해야 한다.
- PPP지원기능은 Linux핵심부로 콤파일되여야 한다.
- PPP쏘프트웨어가 설치되여야 한다.

모뎀

PPP는 전화접속방식용으로 설계되기때문에 모뎀은 PPP의 필수적인 부분이다.

제18장 《주변장치의 리용》에서 이미 모뎀을 어떻게 설치하고 어떤 방법으로 검사하며 어떻게 작업하는가를 고찰하였다. 모뎀을 설치하고 실행시키며 모든것이 순서대로 있는가를 확인하는 검사접속을 만들기 위하여 제18장을 참고할수 있다.

PPP를 구성하려면 모뎀접속속도를 알아야 하며 Linux에서 어느 장치가 리용되는가 하는것도 알아야 한다(/dev/모뎀 혹은 /dev/ttyS0 ~ /dev/ttyS3중의 하나).

핵심부에서의 PPP

Linux핵심부는 높은 유연성을 가지고 설계되였다. 그것은 마우스설정으로부터 PPP와 같은 콤퓨터망설정에 이르는 많은 수법들을 포함(혹은 제외)할수 있다.

Linux로 PPP를 접속하기 위해서는 핵심부가 PPP를 지원할수 있어야 한다. 즉 조작체계가 초기기동하는 동안에 화면에 나타나는 통보문들을 보면서 검사할수 있다. 이와 같은 지령렬을 보면 다음과 같다.

```
PPP generic driver version 2.4.0
PPP Deflate Compression module registered
```

PPP BSD Compression module registered

이때 PPP는 핵심부에서 콤파일된다. 만일 통보문흐름이 너무 빠르다면 dmesg 지령을 리용하여 PPP통문들의 시작부분을 볼수 있다. 즉

```
$ dmesg | less
```

주의 만일 PPP지원을 포함하는 핵심부를 재콤파일하려면 제20장《Linux핵심부의 재콤파일》를 참고할수 있다. 이 장에서는 핵심부를 재콤파일하는데 필요한 단계들을 본다.

PPP쏘프트웨어의 설치

Red Hat Linux 7.1은 완전설치 혹은 기정설치로 PPP쏘프트웨어를 설치한다.

PPP접속을 설치하기 위하여 두개의 프로그람 즉 /usr/sbin/pppd와 /usr/sbin/chat을 리용한다. Red Hat Linux 7.1에서 이것들은 ppp-2.4.0-2프로그람묶음의 부분이며 rpm지령을 리용하여 이것들이 설치되였는가를 볼수 있다. 즉

```
$ rpm -q ppp
ppp-2.4.0-2
```

pppd나 chat가 원만히 설치되지 않았다면 작업을 계속하기전에 PPP쏘프트웨어를 새롭게 설치해야 한다. Red Hat CD-ROM을 /mnt/cdrom에 설치한 다음 ppp-2.4.0-2.i386.rpm프로그람묶음을 다음지령을 실행하여 설치해야 한다.

```
$ rpm -i /mnt/cdrom/RedHat/RPMS/ppp-2.4.0-2.i386.rpm
```

한편 ftp://ftp.samba.org/pub/ppp/에서 Samba download mirror싸이트의 pppd 홈페지로부터 PPP의 최신원천코드들을 내리적재할수 있다. PPP현재 판본은 2.4.1이고 파일이름은 ppp-2.4.1.tar.gz이다.

다음과 같은 지령으로 /tmp와 같은 위치에 있는 문서고를 확장해야 한다.

```
$ tar xzvf ppp-2.4.1.tar.gz
```

그리고 README.linux파일을 주의 깊게 읽어야 한다. 새로운 PPP프로그람묶음설치는 쏘프트웨어의 콤파일뿐아니라 Linux핵심부원천파일들의 개량과 설치된 PPP쏘프트웨어판본에 대응한 Linux핵심부의 재콤파일도 포괄한다.

이 과정에 대한 상세한 정보에 대해서는 모든 장들에서 취급한다. 그러므로 그것은 문서에 가장 많이 남아 있다.

일러두기 내리적재한 Linux 쏘프트웨어에 대한 광범한 원천프로그람들은 http://www-tucows.com에 있는 Tucows Web싸이트와 배포물을 비롯한 수많은 Web싸이트에서 볼수 있다. rpm프로그람묶음을 참고하면 이 프로그람묶음의 많은 원천프로그람들이 http://www.rpmfind.net에 있는것을 볼수 있다.

도형PPP접속

PPP접속을 지원하는 두개의 주요도구들이 있다. 그것들은 Linux콤퓨터를 인터네트에 접속하기 위해서 설계되였다. 즉 Red Hat Linux 7.1에서 GNOME의 RP3과 KDE.RP3의 KPPP는 사용자들에게 될수록 간단한 대면부를 제공한다. KPPP는 인터네트접속의 매 부분을 구성하게 한다.

다음의 절들에서는 몇개의 특별한 설정들을 요구하는 기초적인 인터네트접속을 고찰한다. 그러나 일부 ISP들은 필요한 추가적인 특수설정들을 진행한다. RP3과 KPPP는 진단검색과 특별한 설정을 방조할수 있다. 이 선택적인 설정들에 대한 보다 구체적인 정보는 후에 pppd와 minicom 에 대한 절들에서 볼수 있다.

RP3

RP3은 Red Hat 의 도형PPP관리도구이다. 만일 Linux접속을 조종할수 있는 ISP가 있다면 간단히 설정할수 있는 세 단계들을 거쳐서 인터네트에 접속할수 있다. 후에 이에 대한 구성을 더 정확하게 할수 있다. 즉 접속할 때 제기되는 문제들을 처리하는 말단들을 설정할수 있다. 그러나 RP3은 리용하기 쉽다. 먼저 뿌리사용자로 그것을 구성하고 다음에 체계상에 있는 정해 진 사용자가 인터네트에 접속할수 있다.

RP3구성

기정으로 호출아이콘은 탁상의 오른쪽에 구성된다. RP3을 실행하려면 탁상에 있는 Dialup Configuration아이콘을 두번 찰칵하거나 간단하게 X Window의 말단으로부터 /usr/bin/rp3-config지령을 실행하여야 한다.

재촉문이 나타나면 뿌리통과암호를 입력해야 한다. 만일 RP3 을 처음으로 리용한다면 Add New Internet Connection Wizard가 기동한다. 계속하려면 Next 를 찰칵해야 한다. 아직 모뎀이 구성되지 않았다면 Select Modem 대화칸이 나타나게 된다. 모뎀을 전화선에 련결해야 한다. 계속하려면 Next 를 찰칵해야 한다. 그림 22-1에서 볼수 있는바와 같이 RP3은 모뎀에 접속할수 있는 모든 장치를 검색한다.

일러두기	만일 Add New Internet Connetion Wizard가 나타나지 않고 Internet Connection 대화칸이 나타나면 Add단추를 찰칵해야 한다. 한편 RP3프로그람묶음이 설치되지 않을수도 있다.

만일RP3 이 후보장치를 발견하면 자동적으로 전화소리를 검색한다. 그것은 매 장치파일을 실행한후 그림 22-2에서 보여 준것과 같은 결과를 준다.

만일 모뎀을 발견하지 못하면 다음 보게 될 창문은 그림 22-3과 같다. 이 화면은 여러가지 설정들을 편집하는데 이 설정들은 Linux가 모뎀을 발견해서 리용하도록 한다. 설사 Linux 가 모뎀을 발견했다고 해도 그림 22-2에서 보여 준 Modify This Modem Manually선택항목을 선택하여 이 창문을 계속 호출할수 있다.

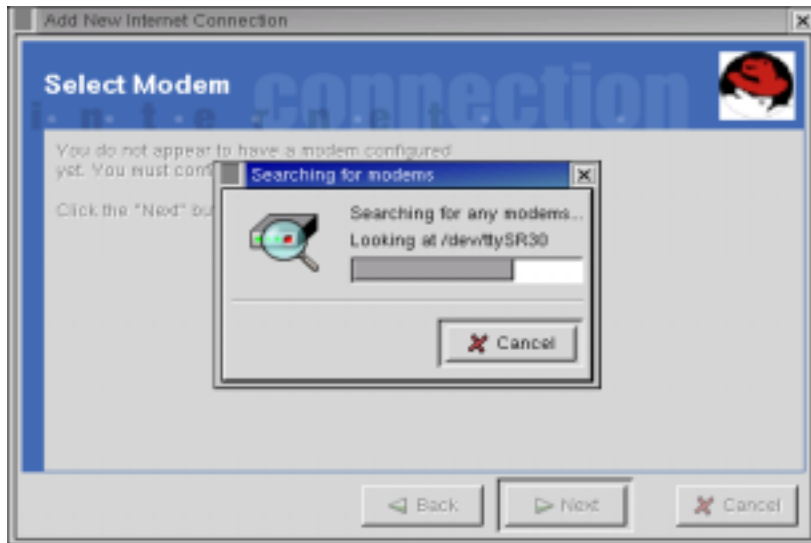



그림 22-1. 모뎀검색

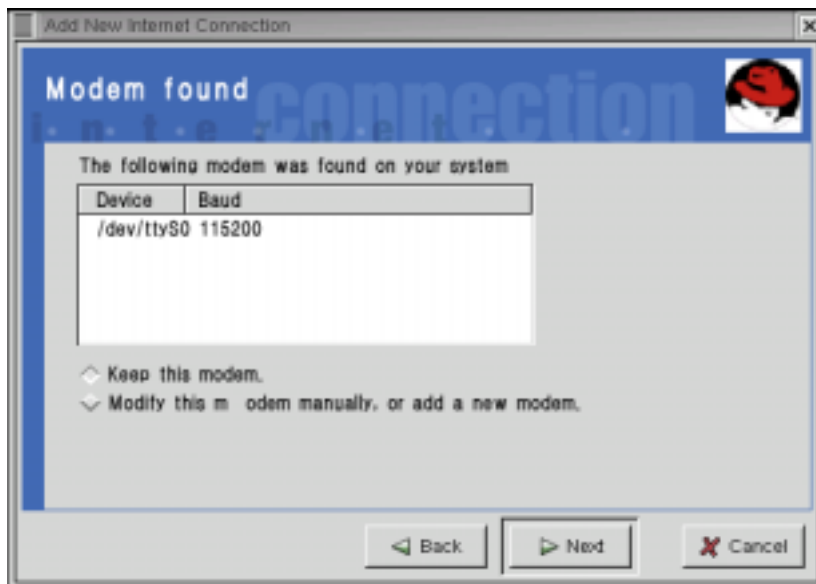



그림22-2. RP3은 한개의 모뎀을 발견한다.

주의

보드속도는 모뎀의 속도와 같지 않다. 그것은 대체로 115,200보드이상으로서 모뎀속도의 4배이다. 그러나 외부에서 들어 오는 잡음으로 인한 자료손실을 줄이기 위해 더 낮은 속도를 설정할수도 있다.

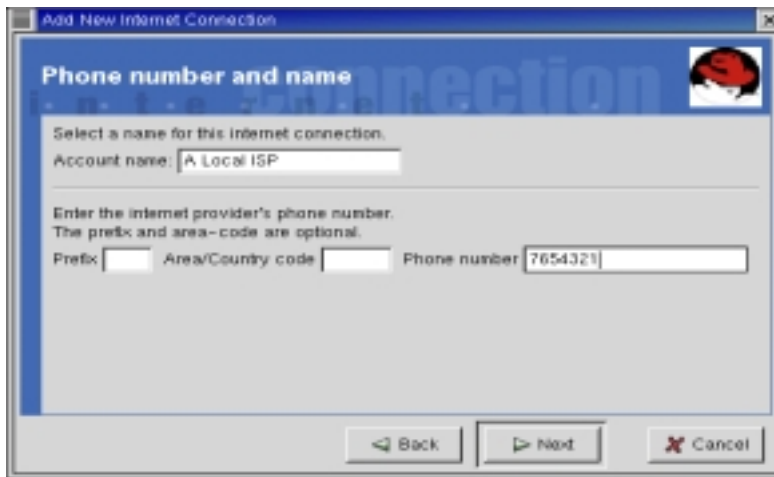



그림 22-3. 수동적인 모뎀구성

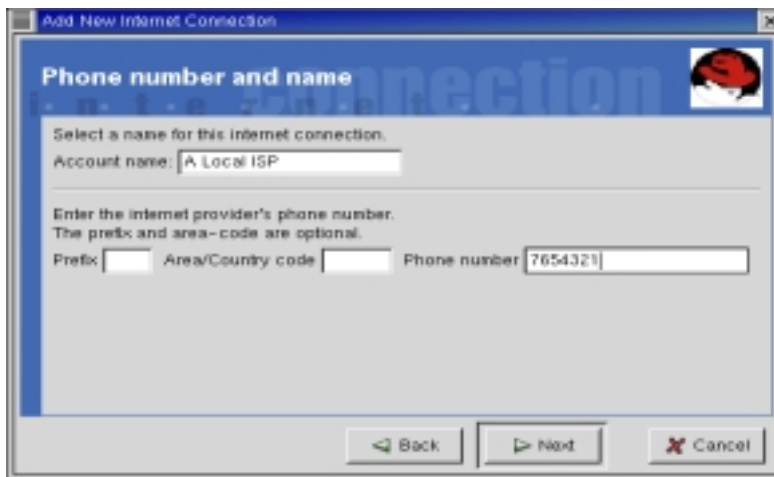



그림 22-4. ISP호출정보의 입력

일단 모뎀이 구성되였으면 Next 를 찰칵해야 한다. 다음단계는 그림 22-4에서 보여준것처럼 기초적인 인터네트접속의 기초들을 설정하는것이다.

- Account Name은 ISP의 이름이다.
- Prefix는 외부호출을 위한 전화번호이다. 앞머리수자가 요구되지 않으면 그곳에 공백을 남겨 둔다.
- Area/Country Code는 ISP호출번호를 가진 국부전화교환기를 호출하는데 필요한 번호이다. 국부전화호출을 통하여 ISP와 결합될수 있다면 그 내용을 비울수 있는데 이 호출은 국부전화회사가 설정한 규정에 따른다.
- Phone Number는 ISP에 대한 국부전화호출번호이다.

Next를 찰칵하고 사용자이름과 ISP에 가입하는데 리용하는 통과암호를 입력하고 다시 Next를 찰칵한다. Other Option창문에서Normal ISP혹은AT&T Global Network

Services를 선택하고 Next를 찰칵한다. 다음창문에서 Finish를 찰칵하면 접속준비가 끝난다

RP3리용

일단 PPP접속이 구성되면 정규적인 사용자들은 체계상에서 그것을 실행할수 있다. GNOME Main Menu 단추로 Programs -> Internet -> RHPPP Dialer를 선택한다. 그림 22-5에서 보여 준바와 같이 Choose창문을 연다. ISP에 할당된 이름을 선택하고 OK를 찰칵하면 Linux는 ISP를 통해 인터네트에 접속된다.

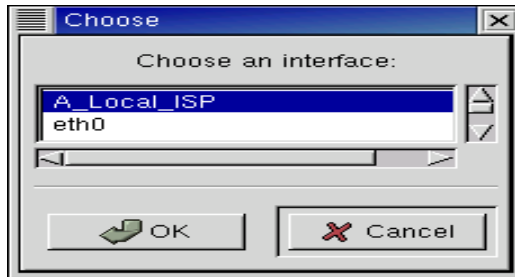



그림 22-5. ISP의 선택

KPPP

인터네트를 호출하는 다른 주요한 Linux GUI방법은 KPPP, KDE PPP Dialer를 리용하는것이다. 만일 KDE탁상에서 작업하면 KDE Main Menu 를 찰칵하고 다음 Intenet -> Internet Dialer를 선택해야 한다. GNOME와 같은 기타 탁상에서 지령창문을 열고 /usr/bin/kppp지령을 입력한다. 그러면 그림 22-6과 같은 KPPP창문이 나타난다.

이 창문에서 SetUp를 찰칵한다. KPPP구성창문이 현시되면 Accounts표쪽을 선택하고 그다음 New를 찰칵한다. 그러면 Create A New Account 창문이 현시되는데 그것은 유럽의 ISP들에서 처음부터 전용화되였다. 만일 유럽이나 뉴질랜드에서 ISP를 리용하지 않는다면 Dialog Setup을 찰칵해야 한다. 그러면 그림 22-7에서 보여 준것처럼 New Account창문이 나타나는데 거기서 ISP와의 접속을 설정할수 있다. 대부분 경우에 그림 22-7에서 보여 준 항목들만을 설정한다.

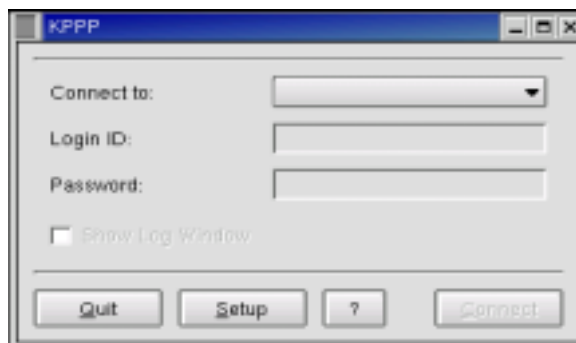



그림 22-6. KDE PPP구성도구

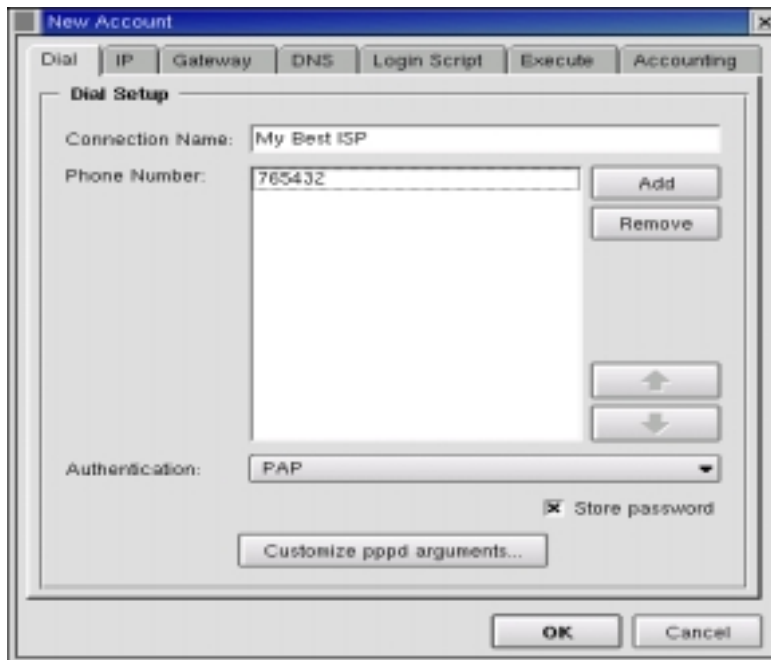



그림 22-7. ISP의 설정

만일 접속에서 문제가 생기면 이 창문으로 돌아 와야 한다. 그림 22-7에서 보여 준 바와 같이 다른 표쪽들로 다음과 같은 선택항목을 구성할수 있다.

Dial 접속할 때 전화번호외에 여러 곳에서 실행할수 있는 프로그람뿐아니라 통과암호인증의 형태도 구성할수 있다.

IP ISP가 정적 IP주소를 할당하면 그것을 여기에 입력한다.

Gateway ISP가 망의 관문를 인터네트에 자동적으로 설정하지 못하면 그와 련관된 관문 IP주소를 여기에 입력할수 있다.

DNS ISP가 접속에서 DNS봉사기들을 자동적으로 설정하지 않으면 그와 련관된 DNS IP주소들을 여기에 입력할수 있다.

Login Script ISP가 가입스크립트를 리용할것을 요구하면 그와 련관된 지령을 여기에 입력할수 있다.

Execute 이것은 접속처리의 여러 단계에서 실행하는 프로그람이나 지령들을 설정하도록 한다.

Accounting 만일 분당 국부전화호출값을 물어야 한다면 여기에 그와 련관된 회계규칙을 입력할수 있다.

일단 필요한 모든것이 구성되면 OK를 찰칵하여 그림 22-8에서 보여 준 KPPP Configuration창문에로 돌아 갈수 있다.

주의 미국의 읽기도구들은 많은 나라들의 전화사용료금을 알아야 한다. 국부호출들은 분당 변경되는데 이것은 인터네트호출에 대한 각이한 요구들을 만든다.

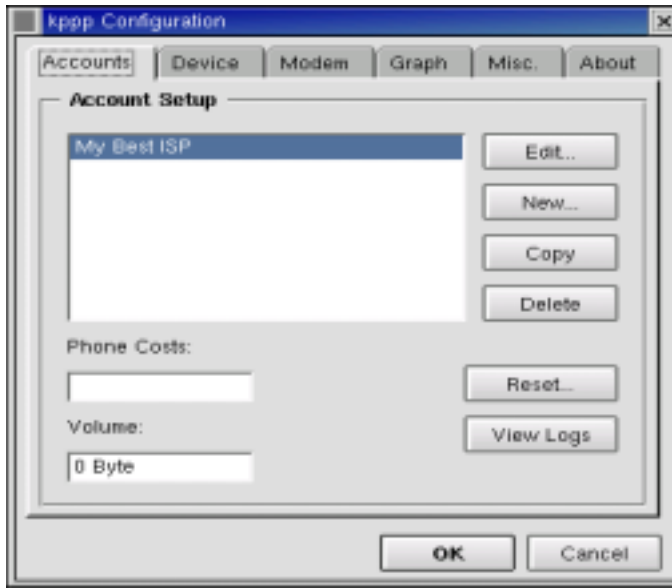



그림 22-8. KPPP구성선택항목

만일 모뎀설정에 문제가 있다면 이 창문으로 돌아 와야 한다. 다음과 같은 각이한 표쪽을 리용하여 많은 방조도구를 리용할수 있다.

Accounts ISP 회계이름을 보여 주는것외에 이 표쪽을 log 파일을 얻을수 있게 한다. 이 파일에는 오유가 발생했을 때 오유가 발생하기전 상태를 회복하는데 필요한 통보문들이 있다.

Device 모뎀과 관련된 하드웨어파라메터들을 설정한다.

Modem 여러가지 방법으로 모뎀을 검사하거나 호출하도록 한다.

Misc 접속과 비접속할 때 혹은 X Window창을 닫을 때 PPP데몬의 상태를 결정한다.

만일 정규사용자들이 호출할수 있다면 suid허가를 설정할수 있는데 이것은 매 사용자가 완전히 호출하지 않고도 다른 프로그람을 실행하게 한다. 다음과 같은 파일에 대하여 허가를 변화시키면 모든 정규사용자들은 /usr/sbin/kppp지령을 실행하여 편의프로그람들을 호출할수 있다.

```
# chmod u+s /usr/sbin/kppp
```

수동적인 PPP접속

RP3과 KPPP편의프로그람들은 대다수의 사용자들에게 아주 쓸모 있다. 만일 의문이 있거나 Linux에서 PPP접속의 내부동작을 리해하고 싶다면 PPP데몬과 다른 접속편의프로그람들이 어떻게 작업하는가를 리해하는것이 중요하다.

Linux에서 PPP접속은 PPP로 구성되고 유지된다. 그러나 pppd는 이미 모뎀과 ISP의 모뎀사이에서 접속이 이루어 지고 필요한 모든 가입이 진행되며 ISP의 체계가 우에서와 같은 모뎀접속방법으로 PPP접속을 만들려고 하고 있다는것을 인정한다.

이것은 마치 제한조건이 많은것처럼 보이지만 실지로는 그리 나쁜것이 아니다. 거의

모든 ISP에서 접속처리과정은 다음과 같다.

1. 모뎀접속이 이루어 진다.
2. 가입과정이 일어 난다.
3. 일단 사용자가 확증되면 ISP는 말단IP주소를 말단체계에 보냄으로써 ppp접속을 진행한다.

pppd 프로그람은 먼저 1, 2단계를 진행하고 그에 기초하여 3단계를 진행하게 한다.

이 처리를 할수 있는 준비가 되면 pppd를 실행하기전에 먼저ISP에 모뎀을 접속하여야 한다. 일단 접속이 이루어 지면 pppd로 PPP 접속을 할수 있다.

PPP접속만들기

pppd 쏘프트웨어는 도구를 집합시키고 전체 접속과정을 조종하는 chat 프로그람으로 작업하기 위해서 설계된다. 접속하기전에 일부 정보를 수집해야 한다.

- 접속하는 IP주소를 매번 할당하였는가 혹은 할당된 IP주소를 변함없이 가지고 있는가? 대부분 ISP들이 접속할 때 동적IP주소를 할당한 기초우에서 작업하기 때문에 역시 이에 기초하여 작업해야 한다.
- ISP에 어떻게 가입하는가? 한가지 방법은 순수한 본문재촉문 및 응답가입방식들을 리용하는것이다 . 다른 하나는 PAP라고 하는 특수한 인증규약을 리용하는것이다. 기타 방법은 CHAP로 알려 진 규약을 리용하는것이다. 만일 minicom이나 다른 말단쏘프트웨어나 어떤 가입재촉문상에서 ISP에 접속하려면 순수한 본문재촉기호로 가입해야 한다. 만일 접속후 문자들이 무질서하게 나타나거나 혹은 전혀 나타나지 않는다면 PAP나 CHAP인증을 리용해야 한다.

주의	이 절의 모든 실례들은 콤퓨터에서 뿌리사용자로 작업하는것을 전제로 한다. PPP접속과정이 대면부조작과 망경로만들기와 삭제를 요구하기때문에 관련된 구성프로그람은 뿌리사용자에 의하여 실행되여야 한다.

일러두기	규약뿐만아니라 특별한 IP주소에 대한 정보도 RP3이나 KPPP구성을 지원할수 있다. 만일 그에 대한 확신이 없거나 이 정보가 없으면 ISP에 문의해야한다.

순수한 본문재촉기호와 접속

더 작업하기가 쉬운 순수한 본문재촉기호를 보자.

pppd는 자체로 모뎀장치를 조종할수 있지만 실제적인 대화나 가입을 진행할수 없기때문에 모뎀에 지령을 주고 필요한 가입정보를 제공하여야 한다. chat프로그람으로 이 작업을 진행할수 있다. chat프로그람이 있으면 대화교환을 만들수 있다.

실례로 모뎀에서 표준대화과정에 빈 말단창문에서 말단편집프로그람을 실행하고 **ATDT1234567**과 같은 전화지령을 입력할수 있다. 이에 대한 해답은 보통 CONNECT 115200과 같은 접속문자렬인데 사용자는 그에 응답하지 않는다.

주의	만일 **ATDT**지령에 대한 응답이 필요하면 모뎀을 접속해야 한다. 말단접속을 조종할수 있는 ISP나 기타 봉사기전화번호를 **1234567**로 바꾼다.

이 변환을 간단한 chat 스크립트로 진행할수 있다. 즉

"" ATDT1234567 CONNECT ""

이 스크립트는 두개의*expect-send pairs* 즉 "" ATDT1234567 와 CONNECT ""로 이루어 져 있다. expect-send pair는 공백으로 분리된 2개의 정보를 포함한다. 첫 부분에는 chat가 기대하는것이 아무것도 없으며(첫번째 부분의 두개의 인용부호들) 문자렬 ATDT1234567를 보내여 응답한다. 다시말하여 스크립트가 처리를 시작하자마자 ATDT 지령은 선택된 전화번호를 돌린다. 다음에 두번째 부분이 처리되고 chat는 문자렬 CONNECT를 요구하며 아무런 처리도 하지 않는것으로 응답한다(두번째 부분의 두개의 인용부호). 만일 이것이 복잡한 스크립트라면 chat는 이 시점에서 완료하고 끝낸다.

물론 chat 스크립트가 좀 더 길어야 한다. chat스크립트를 완성하려면 즉시에 가입대화조종이 어떻게 보이는가를 조사해야 한다. 이 작업은 minicom과 같은 정규말단쏘프트웨어로 가입함으로써 진행할수 있다.

대부분의 ISP들에서 가입재촉문들은 일반적으로 다음과 같은 형태를 가진다.

Username:
Password:

혹은

Login:
Password:

또는

ogin:
ssword:

이 장에서는 실례로 첫번째 경우를 고찰한다. 만일 가입재촉문이 다르면 작업할 때 적당히 바꾸어야 한다.

주의	Login:와 Password:대신에 여러가지 조작체계들에 맞게 일부ISP들은 콤퓨터에 ogin:과 ssword:혹은 sername:와 ssword:를 전송하도록 설정된다.

그러면 다음에 필요한 전송쌍은 무엇인가? 접속후 Username:재촉문이 현시된다. 여기에 사용자이름(사용자이름이 testuser라고 하자)을 주는것으로 응답한다. 호상작용하는 전송쌍은 Username: *testuser*로 될수 있다.

일단 사용자이름이 주어 지면 Password: 재촉문이 나타나는데 여기에 통과암호를 준다(이것을 testpassword라고 하자). 이것은 전송쌍 Password:*testpassword*를 만든다.

많은 ISP들에서 이것은 충분히 가능하며 PPP는 정확한 통과암호문이 입력된 다음에야 ISP의 체계에서 작업할수 있다. 규모가 더 작은 체계들에서 ISP 의 콤퓨터들은 지령재촉문을 제시하는데 여기에서 PPP를 실행하기 위해 한개의 지령을 입력해야 한다. 이 경우에 보충적인 쌍을 만들어야 한다.

실례로 다음과 같은 완전한 chat스크립트를 볼수 있다.

"" ATDT1234567 CONNECT ""Username: testuser Password: *testpassword*

chat프로그람(이것은 보통 /usr/sbin에 있다.)으로 이것을 리용하기 위해서는 간단하게 스크립트를 chat의 인수로 넘겨야 한다. 즉

/user/sbin/chat "" ATDT1234567CONNECT "" Username: testuser Password: *testpassword*

그리고 지령행에 이것을 써넣지 않으려면 Enter를 눌러야 한다. pppd로 통합되거나 그 프로그람으로 모뎀을 호출하지 않고도 chat는 쉘을 리용하여 chat하려고 한다. 스크립트를 검사하기 위해 이것을 리용할수 있다. 지령재촉문에 지령을 간단히 입력하면 된다. ATDT1234567가 나타나면 **CONNECT**를 먼저 입력하고 다음에 **Username:**를 입력한다. 이에 대한 응답으로 *testuser* 가 나타나야 한다. **Password:**를 입력하면 이에 대한 응답으로 *testpassword*가 나타난다.

주의	만일 chat스크립트로 문제를 해결하려면 chat지령에 –v기발을 추가해야 한다. 이때 체계로그에 스크립트의 출구를 전송하는데 여기서 문제풀이결과를 분석할수 있다.

작업스크립트가 있으면 pppd로 그것을 통합하여야 한다. 이를 위하여 pppd선택항목의 일부를 리용한다(pppd응용프로그람들은 많은 추가적인 선택항목들을 가지는데 이 항목들은 pppd사용설명서페지에서 볼수 있다.). 선택항목들은 다음과 같다.

Connect 이 항목은 쓸수 있는 전화선에 접속하는데 리용되는 프로그람이나 지령을 정의하기 위해 비여 있다. 이 경우에 chat 프로그람과 그의 스크립트를 정의하기 위해 connect항목을 리용한다.

Noipdefault pppd의 기정행동규범은 주콤퓨터이름에 따라 국부콤퓨터의 IP주소를 결정하는것이다. 그러나 ISP가 동적IP주소를 할당한다면 이 기능은 더 일반적이며 noipdefault 항목을 리용하여 접속된 원격조종콤퓨터로부터 IP주소를 얻도록 pppd와 대화하는데 리용된다.

Defaultroute 이 항목은 기정판문로서 원격조종체계(ISP)를 리용하여 체계의 경로표에 기정경로를 추가하도록 pppd와 대화한다. 이 항목은 접속이 파괴될 때 제거된다.

pppd지령구조는 다음과 같다.

$ pppd *device speed action chatcommands pppdoptions*

만일 장치 /dev/modem과 접속되여 있다면 모뎀은 115200bps의 높은 속도를 가지는데(이 경우에 56kbps 모뎀이다.) 이때 다음과 같은 지령을 리용하여 chat스크립트를 접속할수 있다.

```
$ pppd /dev/modem 115200 connect '/usr/sbin/chat "" ATDT1234567  CONNECT `
"" Username: testuser Password: testpassword' noipdefault defaultroute
```

이 지령이 실행되면 모뎀은 번호를 돌려 접속되며 인증확인을 거쳐 PPP의 접속을 확증한다. 이 과정이 성과적으로 진행되면 두가지 현상이 일어 난다. 즉

첫째로, 기발이나 인수들이 없이 지령 /sbin/ifconfig를 전송하려면 이 지령에서 류

사한 PPP대면부를 포함한 대면부의 목록을 돌려 주어야 한다.

```
ppp0      Link encap:Point-to-Point Protocol
          inet addr:194.209.60.101 P-t-P:194.209.60.97 Mask:255.255.255
          UP POINTTOPOINT RUNNING MTU:1500 Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0
          TX packets:11 errors:0 dropped:0 overruns:0
```

둘째로, 항목들을 경로표에 첨가하여 원격조종콤퓨터를 통한 기정경로를 만들어야 한다(기발이나 인수들이 없이 /sbin/route 지령을 리용하여 검사한다.).

```
Kernel IP routing table
Destination Gateway       Genmask          Flags Metric Ref  Use Iface
du1.paus.ch *             255.255.255.255  UH        0     0    0  ppp0
default                   du1.paus.ch 0    UG        0     0    0  ppp0
```

이 실례에서 du1.paus.ch는 PPP로 접속되는 원격조종콤퓨터이며 기정관문으로 봉사하고 있다.

PAP인증으로 접속

PAP인증으로 접속하는 방법은 본문재촉문실례에서와 같은 기초적인 원리에 따른다. 여기서 사용자이름과 통과암호를 제공하는 방법이 차이난다.

첫번째 기본차이점은 PAP스크립트의 부분으로서 지령행에서 사용자이름과 통과암호를 제공하지 않는것이다. 대신에 PAP인증처리에 리용되는 특별한 secrets파일에서 항목들을 작성한다. Red Hat Linux 7.1에서 이 파일은 /ect/ppp/pap-secrets/이다. 기정으로 이 파일은 뿌리사용자에 의해서만 읽을수 있고 쓸수 있다. 다른 사용자들은 이 파일에서 읽기호출을 하지 못한다.

secrets파일에서 입력지령은 일반적으로 다음과 같다.

username servername password

그러나 전화모뎀을 통하여 봉사기에 접속하였으므로 두번째 입력은 필요 없다. 실례로 만일 앞의 절에서 리용한 검사용사용자이름과 통과암호를 리용한다면 입력은 다음과 같다.

testuser * testpassword

별표는 이 통과암호가 임의의 대면부에서 접속하는데 리용될수 있다는것을 표시한다.

일단 사용자이름과 통과암호가 파일에 있다면 새로운 chat 스크립트를 작성해야 한다. 대부분 ISP들에 대하여 모뎀접속이 확립되자마자 PAP인증을 시작할수 있다. 따라서 chat스크립트는 더 간단해 진다. 즉

"" ATDT12345678 CONNECT ""

이 스크립트는 ISP를 불러 낸 다음 chat프로그람을 끝내고 인증처리에 들어 가기전에 접속통보문이 접수되는가를 확인한다.

끝으로 pppd:user 를 위한 선택항목을 한개이상 소개해야 한다. user 항목은 pap-secrets 파일로부터 인증되는 PAP사용자를 표시한다. 즉 마지막으로 다음의 pppd

지령을 입력한다.

```
$ pppd /dev/modem 57600 connect '/usr/sbin/chat "" ATDT1234567 CONNECT' noipdefault defaultroute user testuser
```

초기본문가입실례에서와 같이 모든것들이 ifconfig와 route를 차례로 리용하고 있는가를 조사할수 있다.

접속이 있을 때

인터네트와 접속하고 충분히 인터네트에 접속할수 있는가를 확인하여야 한다. 이를 위해서 DNS봉사기들이 ISP의 이름봉사기를 가리키도록 구성되였는가 확인해야 한다.

이 조작은 두개의 파일 즉 /etc/host.conf와 /etc/resolv.conf를 편집하여 진행할수 있다. 이 두 파일에 대해서는 제28장《Ethernet 망을 위한 Red Hat Linux 7.1의 구성》에서 더 상세하게 고찰한다. 이 절에서는 빨리 직결하기 위해서 필요한 기초적인 파일들을 고찰한다.

파일 /etc/host.conf는 다음과 같은 지령들을 포함한다.

```
order hosts,bind
multi on
```

다음 /etc/resolv.conf는 적어도 두개의 지령들을 포함한다. 즉

```
search
nameserver 100.100.100.100
```

이 경우에 역시 IP주소 100.100.100.100을 이름봉사기로 알려 진 ISP의 DNS봉사기 IP주소로 교체한다. ISP는 이 정보를 제공할수 있다. 만일 ISP가 하나이상의 이름봉사기로 제공된다면 다음과 같이 매 행들에 대해서 개별적인 행들을 만들수 있다.

```
search
nameserver 100.100.100. 100
nameserver 200.200.200.200
```

일단 이 과정이 끝나면 이름을 결정하고 인터네트를 호출할수 있어야 한다. 또한 Netscape 와 FTP 와 같은 인터네트쏘프트웨어도 실행할수 있어야 한다. 이 프로그람들은 제23장 《World Wide Web의 리용》에서 보게 된다.

정지

인터네트의 리용이 완성되였으면 전화비용과 직결상태를 줄이기 위해서 정지하여야 한다. 이를 위해 pppd 처리를 제거해야 한다.

뿌리사용자는 ps지령을 리용하여 pppd프로그람의 ID처리를 발견할수 있다. 즉

```
$ ps x | grep pppd
```

pppd에 대한 입력은 다음과 같다.

```
1316  ttyS0 S 0:00 /usr/sbin/pppd /dev/modem 115200 connect /usr/sbin/chat "" AT
```

첫번째 수는 프로쎄스ID이다. 이 경우에 kill지령으로 이 프로쎄스를 제거할수 있다.

즉

$ kill 1316

pppd가 제거되면 모뎀도 정지해야 한다.

수동적인 인터네트접속의 자동화

PPP로 인터네트와 접속할수 있다 하더라도 긴 pppd와 chat 지령들을 입력해야 한다면 인터네트에 접속한 Linux의 리용은 실제적으로 제한을 받는다.

이러한 제한성을 개선하기 위하여 두개의 스크립트를 작성할수 있다. 하나는 전화를 위한것이고 다른 하나는 정지를 위한것이다. 이러한 두개의 스크립트들을 dial 파 hangup으로 부르기로 한다(론리적선택들, yes?). /usr/local/bin과 같은 경로상의 등록부에 이 파일들을 배치하려고 한다.

아직 스크립트에 대한 파악이 없기때문에 이 두 스크립트들은 스크립트들의 리용에 대한것을 정확히 내리적재할수 있도록 하는 간단한 론의를 제시한다. 이 스크립트들은 임의의 본문편집프로그람으로 작성될수 있다. 여기서 한행으로 나타나는 행들이 파일에 한개 행으로 있는가를 확인하는것이 중요하다.

dial스크립트에서 초기PAP실례가 인터네트에 접속하는 통로라고 하자.

일단 스크립트를 작성하였으면 chmod지령으로 실행할수있다. 즉

$ chmod 700 dial hangup

허가700은 뿌리사용자와 그밖의 다른 사용자가 대본을 읽을수도 있고 쓸수도 있으며 실행할수 있게 한다. 만일 이와 같은 그룹에서 다른것으로 스크립트를 리용할수 있다면 700을 750으로 바꾸어야 한다. 만일 Linux체계를 호출하는 사용자가 스크립트를 리용하게 하려면 700대신에 750을 입력해야 한다.

일러두기 처음부터 이 스크립트를 작성하지 말아야 한다. 만일 설치된 ppp rpm프로그람묶음이 있으면 이미 구성된 스크립트들은 **/usr/share/doc/ppp-2.4.0/scripts**등록부에 있는 Red Hat Linux7.1에서 리용할수 있다. 실례로 아래에서 보게 되는 **dial**과 **hangup**스크립트들과 같이 **ppp-on**과 **ppp-off**스크립트들을 설정할수 있다.

dial스크립트

dial스크립트는 다음과 같다.

```
#!/bin/sh
/usr/sbin/pppd/dev/modem 115200 connect \
'/usr/sbin/chat "" ATDT1234567 connect' \
noipdefault defaultroute user testuser
```

첫번째 지령은Linux의 /bin/sh등록부에 있는 Bourne Shell로서 스크립트를 처리하도록 알려 주는 지령이다. 다른 지령들은 이미 앞에서 설명하였다. ISP의 수자가 1234567로 교체되였는가를 확인한다. 필요하다면 모뎀장치파일을 /dev/modem으로 바꾸고 모뎀속도를 115200으로 변환시켜야 한다.

pppd지령은 세개의 행으로 이루어 진 한개의 지령이라는데 주목을 돌려야 한다. 이 지령의 첫번째와 두번째 행의 끝에 있는 \ 는 아직 지령들이 끝나지 않았으며 지령이 다음행에서 계속된다는것을 표시한다. 이것은 스크립트를 한눈에 볼수 있게 해준다.

hangup스크립트

hangup스크립트는 다음과 같다.

```
#!/bin/sh
kill  'cat /var/run/ppp0.pid'
```

dial스크립트와 같이 이 스크립트도 Bourne Shell로 처리된다는것을 명백히 해두자. 다음 앞서 취급된것과 약간 차이나는 통로의 처리들은 제거한다.

pppd프로쎄스는 대부분의 현대적인 Linux체계상의 /var/run등록부에 있는 한개 파일에 자기의 프로쎄스ID를 기록한다. 파일이름은 확장자가 .pid인 장치이름으로 이루어 진다. 한개의 모뎀만이 있고 한번에 하나의 PPP접속이 진행되면 그 장치를 ppp0이라고 하고 스크립트에 /var/run/ppp0.pid를 직접 써넣을수 있다.

cat지령은 표준출력으로 화면에 ppp0.pid파일의 내용을 간단히 현시한다. kill 지령으로 cat지령(이것은 pppd프로쎄스ID이다.)의 결과를 통과시키기 위해 cat지령을 거꿀인용팔호(`)안에 넣는다.

주의	표준미국식영어건반의 왼쪽 웃구석에 있는 물결표식(~)아래에서 거꿀인용팔호(`)를 볼수 있다.

그러나 다중 PPP대면부들이 있고 어느 하나를 정지시키려면 어떤 현상이 일어 나겠는가? 다음의 지령이 있으면 hangup스크립트를 빨리 수정할수 있다.

```
#!/bin/sh
kill `cat /var/run/$1.pid`
```

여기서 ppp0을 $1로 바꾸었다. $1은 스크립트에서 첫번째 첨수의 값이 이 위치에 놓여야 한다는것을 나타낸다. 이때 스트립트에 대면부이름을 변수로 넣을수 있다. 지령

$ **hangup ppp1**

은 ppp1대면부에서 리용되고 있는 모뎀을 정지시킨다. 첫번째 모뎀은 ppp0대면부에서 리용되고 있기때문에 hangup ppp0 지령을 대신 실행해야 한다는데 주의를 돌려야 한다.

▶미리 알아두기

이 장에서는 Linux체계로 인터네트에 접속하는 수법들을 보았다. 즉 탁상에서 하나의 콤퓨터를 가지고 PPP통신규약을 리용하여 세계의 모든 콤퓨터들과 접속하는 방법을 보았다.

제23장 《World Wide Web의 리용》에서는 이 새로운 기능의 우점을 설명하고 Linux가동환경에 대한 Netscape, 초기Web열람기, 인터네트의뢰기의 리용방법을 배우게 된다.

그다음 제24장 《전자우편의 읽기》에서는 Linux에서 리용할수 있는 풍부하고 다양한 전자우편쏘프트웨어를 보고 전자우편리용에서 자기 요구에 아주 적합한 전자우편쏘프트웨어를 선택할수 있다.

제23장. World Wide Web의 리용

인터네트에 정확히 련결되여 있는 Linux 체계를 리용하는 사용자들이 제일 먼저 하고 싶어 하는것은 WWW를 읽어 보는것이다.

이 장에서는 Linux에서 쓸모 있는 수많은 Web 열람기의 기능들을 설명하고 대표적인 두개의 열람기 즉 Netscape 6 과 Lynx에 대하여 구체적으로 설명한다.

Netscape 6은 Netscape계렬의 무료 Web열람기와 인터네트의뢰기이다. 이것은 Linux를 비롯하여 Windows와 Macintosh, Unix조작체계에서 쓸수 있다.

주의 만일 Netscape Web싸이트로부터 내리적재된 내용을 조사할 때 더 새로운 판본이 요구되면 이 장의 마지막에 서술한 내용에서 그에 대한 소개를 보면 된다.

이 장에서는 우편함을 고찰한다. 대부분 인터네트의 홈사용자들은 인터네트봉사자(ISP)의 봉사기상에서 유지되는 원격우편함들을 가지고 있다. 그러나 그들은 Eudora나 Netscape의 우편읽기장치와 같은것들을 리용하여 우편을 읽을 때 자기의 우편쏘프트웨어로 국부적으로 작성한 우편함에 통보들을 복사한다.

또 한가지 방법은 우편을 직결로 읽는것이다. 이 방법에서는 새로운 우편을 읽을 때마다 국부적인 우편함을 만들지 않고서도 우편쏘프트웨어에 의하여 우편함을 봉사기상에 남겨 놓고 간단하게 조종할수 있다.

이 장에서는 또한 Unix/Linux본문방식열람기들에서 de facto 표준인 Lynx도 취급한다. Lynx는 Netscape보다 력사가 오래며 Web 기술이 간단한 본문방식하이퍼본문체계에서 작은 량의 기술정보들을 전달하는데 리용된다. Lynx는 계속 진화발전하여 왔으며 Linux의 거의 모든 배포물들에 들어 있다. 직결하여 정보들을 검색하고 다시 비직결로 되는 고속적인 방식은 Netscape 6의 환상적인 도형과 아이콘구동대면부를 요구하며 그 기간에는 Lynx를 없앤다.

Linux열람기의 개요

Linux로 작업하는 X Windows환경에는 여러개의 쓸모 있는 열람기가 있다. 이러한 열람기들로는 WWW협회(W3C) 에서 개발한 검사가동환경 Amaya로부터 Netscape와 같은 주류의 열람기들에 이르기까지의 열람기들을 들수 있다. 이 절에서는 이 열람기들을 잠간 소개하고 Linux에서 리용할수 있는 Web 쏘프트웨어의 범위를 확정한다.

Amaya

Amaya는 새로운 Web규약과 자료형태에 대한 검사기로서 W3C가 개발한 Web열람

기이다. W3C는 WWW개발과 보조를 맞추기 위해 Tim Berners-Lee에 의해 창안된 국제조직이다. Berners-Lee가 WWW의 창안자로 자주 서술되는것은 W3C가 이 체계에 대한 새 표준들을 개척하는 부문으로서 자체의 열람기를 개발하였기때문이다.

Amaya는 PNG와 같은 새로운 도형형태에 따르는 계단식용지들과 HTML을 지원하도록 제공된 결합Web열람기 및 색인도구이다. Amaya는 또한 강력한 수학편집기도 가지고 있다.

Amaya가 앞 가장자리(leading-edge)특성들을 제공한다 해도 그것은 Netscape와 같은 업무적인 열람기가 제공한 Web 의 모든 특성들을 다 제공하지는 못한다.

Amaya는 홈페지 http://www.w3.org/Amaya/에서 적재할수 있다. 여기서 쓰는 현재판본은 4.3.2이다. 그림 23-1은 표본적인 Amaya 창문을 보여 준다.

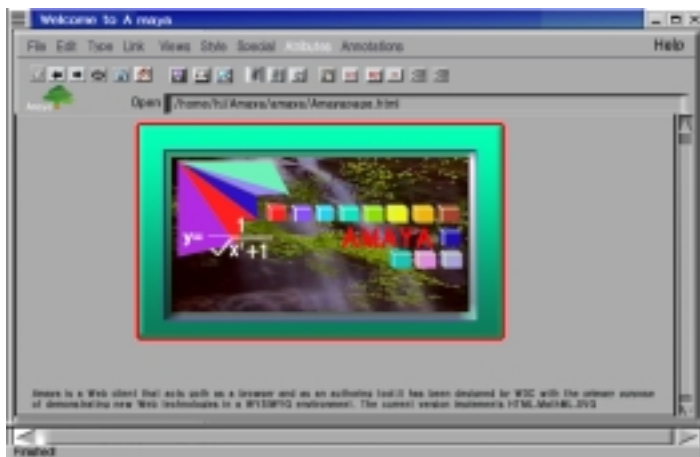



그림 23-1. Amaya Web열람기

Lynx

Lynx는 WWW의 본문열람기에 대한 de facto표준이며 Windows 95이상의 판본, MS-DOS와 OS/2의 각종 판본들을 비롯하여 Linux와 Unix의 각이한 판본들에서 리용할수 있다. Linux홈페지는 http://lynx.browser.org/에서 볼수 있다.

왜 본문열람기를 리용해야 하는지 의문스러울수 있다. 누구에게나 도형열람기를 지원할수 있는 장치들이 있는것은 아니다. 도형을 지원하는 자료접속도 마찬가지이다. 실례로

- 대부분의 공동서고들은 Lynx를 리용한다. Lynx를 호출하는데 필요한것은 dumb말단이다. 제한된 축적을 가진 서고는 간단한 봉사기로부터 실행하는 Lynx를 가진 말단그룹을 설정할수 있다.
- 소규모의 개발령역에서 많은 기능들은 Lynx를 리용한다. 값이 눅은것으로 하여 Lynx는 세계적으로 소규모개발령역에서 대중적으로 리용된다. 이러한 비용으로 하여 이 범위에서 리용되는 콤퓨터들의 능력은 제한되여 있다. 다른 제한으로서는 값이 눅은 전화봉사와 속도가 뜬 모뎀접속이다. Lynx가 도형열람기보다 능력이 작고 속도가 더 낮은 모뎀을 요구하기때문에 이 제한에는 안성맞춤이다.

그러한것으로 하여 Lynx는 프레임을 리용하여 싸이트를 보는 방법을 비롯하여 오늘날 개선된 특성에 대처하는 의미들을 가진 본문열람기에 포함된다. 그림 23-2는 지령행

대면부에서 실행되는 Lynx를 보여 준다. Lynx의 구체적인 응용은 이 장의 뒤에서 서술한다.

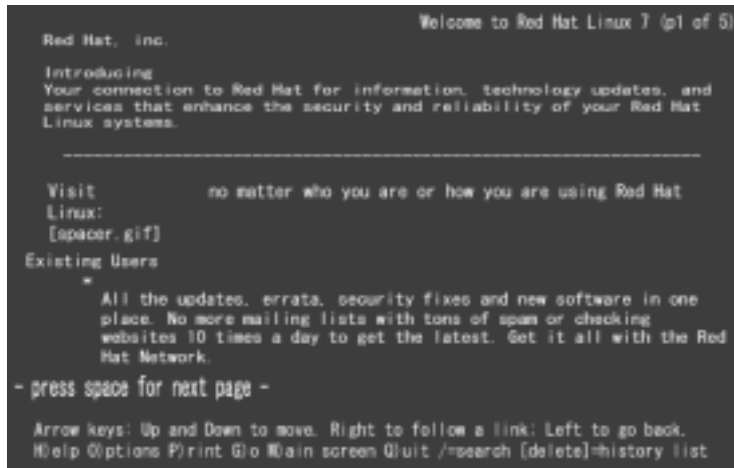



그림 23-2. Lynx Web열람기

Opera

내리적재할수 있는 대부분 Linux 열람기들중의 하나는 그림 23-3에서 보게 되는 Opera이다. 그것은 《작을수록 빠르다》라는 성구로부터 개발되였다. 개발자들은 그 특성페지에서《Opera는 세계의 가장 빠른 열람기로 유명해 지게 될것이다.》라고 진술하였다. 현재준위 Opera 5를 이미 내리적재한 Netscape 6과 비교하면 그것은 1/10크기의 파일로 내리적재될수 있다.

크기는 작지만 거대한 열람기의 기본특성들을 제공한다. 즉

- ICQ표준에로 즉시 통보하기
- 유연한 표현방식으로 개발될수 있는 sheet기능을 계단화하기
- 실험적인 기초에 따르는 무선규약(WAP)지원
- Java

http://www.opera.com의 Web싸이트로부터 Opera를 내리적재할수 있다. 노르웨이에는 현재 내리적재봉사기들만 있기때문에 다른 나라에 대한 내리적재속도는 자주 변할수 있다.

Mosaic

Mosaic는 업무용으로 리용한 첫 도형Web열람기였다. National Center for Supercomputer Applications로 개발된 Mosaic는 1980년대에 나온 열람기인 본문방식 Gopher를 크게 갱신한것이다.

초기에 Mosaic의 판본은 Sun계렬과 같은 대중적인 Unix체계의 X Windows환경에서 리용되였다. WWW의 초기사용자들은 도형과 본문 그리고 마우스로 파일탐색을 쉽게 할수 있는 대면부를 새로 결합한 Mosaic의 페지들에 현혹되였다. Mosaic개발자의 한 사

람인 마크 앤드레슨(Marc Andreesen)은 표준Mosaic인 Mosaic Godzilla 즉 Mozilla에 기초한 Netscape를 계속 개발하고 있었다. 세계적인 범위에서 볼 때 결코 같은것이 있어 본적이 없다.

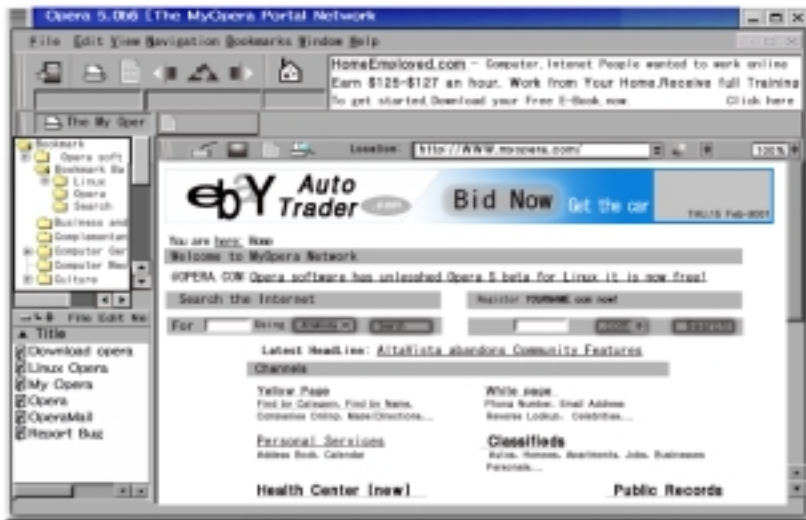



그림 23-3. Opera Web열람기

Mosaic가 실지 개발된 기간은 얼마 안되지만 마지막판본(X Windows인 경우 판본 2.6)은 Urbana-Champaign에 있는 일리노이주종합대학의 Mosaic Web페지 즉 http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/ 에서 리용할수 있다. 그림 23-4는 이미 Web개척자들이 자기들의 열람기로 Mosaic를 리용했던 장면을 보여 준다.

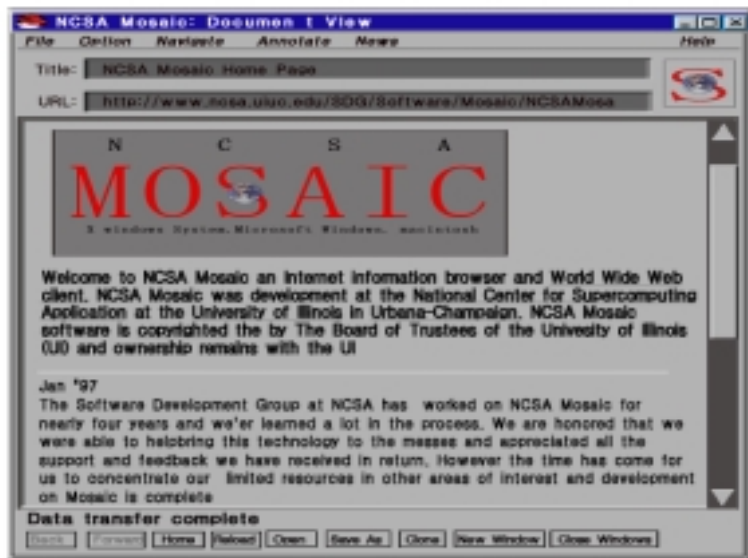



그림 23-4. X Windows에 대한 지난 시기의 Mosaic판본

Netscape 6

Navigator Web열람기를 비롯한 Netscape 6은 WWW에서 가장 기본적인 열람기라고 볼수 있으며 현재 표준이라고 할수 있는 열람기특징들의 원천으로 되였다. 비록 Netscape가 더는 열람기시장경쟁에서 주도권을 얻지 못한다고 해도 Linux에 대한 정합성과 자기원천코드에 대한 공개물들은 여전히 생활력을 가지고 있다. 그래서 Netscape 6은 Linux를 비롯하여 그것을 리용하는 모든 가동환경에서 시종일관 거의 변함 없는 행동규범들을 보여 준다. 그림 23-5는 Linux에서 리용하는 Netscape 6을 보여 준다. 처음에 그것은 Windows사본과 같이 보인다. Linux에 대한 Netscape 6에 대해서는 이 장의 뒤에서 구체적으로 서술하였다.

그림 23-5. Linux를 위한 Netscape 6

Netscape의 설치와 리용

Netscape Communicator판본4.76은 이 책과 함께 제공된 CD-ROM에 들어 있다. Netscape Web싸이트 http://home.netscape.com/ 로부터 Netscape의 판본 6을 내리적재할수 있다. 이 절에서는 Netscape 6에 기본중점을 두고 고찰한다.

이 절에서는 Linux에서 Netscape Communicator와 Netscape 6의 설치에 대하여 잠간 보고 쏘프트웨어를 간단히 소개한다. 이 소개에서는 Netscape를 구체적으로 설명하지 않지만 직결로 충분한 정보를 제공 받을수는 있다. Netscape 6은 Netscape의 기능들을 리용하는데 필요한 모든 정보를 갖추고 있는 완전한 직결봉사기능을 갖추고 있다.

CD-ROM으로 설치

Netscape Communicator 4.76은 Red Hat Linux 7.1의 한 부분이며 대체로 기정으로 설치된다. 만일 Netscape가 체계에 없다면 이 책과 함께 제공되는 CD-ROM으로 설치할수 있다.

Communicator설치를 위하여 먼저 /mnt/cdrom과 같은 론리적인 위치에 CD-ROM을 설치해야 한다. 다음 아래와 같은 Netscape통신묶음을 설치하기 위해 rpm지령을 리용한다.

```
$ rpm - i /mnt/cdrom/RedHat/RPMS/netscape-common-4.76-11.i386.rpm
$ rpm - i /mnt/cdrom/RedHat/RPMS/netscape-communicator-4.76-11.i386.rpm
```

이 지령들은 체계에 완전한 Netscape Communicator판본을 설치한다.

Web으로 설치

만일 Netscape Web싸이트로부터 Netscape의 새로운 판본을 내리적재하려면 설치가 좀 차이난다. 실례로 Netscape-i686-pc-linux-gnu-sea_tar.gz라고 하는 압축된 tar와 같은 Netscape판본6.0을 내리적재할수 있다.

일러두기

이 Netscape 6은 거의 30MB이다. 만일 이 자료를 내리적재하려 하지 않는다면 Netscape 설치기를 내리적재하는데 그것은 인터네트상에서 필요한 프로그람만을 내리적재하여 설치한다. 이를 위해 64KB 파일인 netscape-i686-pc-linux-gnu-installer.tar.gz파일을 내리적재하고 실행한다.

주의

열람, 우편관리, Web페지작성프로그람을 포함하는 Netscape프로그람묶음은 판본이 4.76인 Netscape Communicator로 알려 졌다. Netscape가 새로운 프로그람묶음을 공개할 때 그것은 직접 판본 6.0으로 이행하였으므로 판본 5.0은 없다.

/tmp/와 같은 림시장소에 이 서고를 적재해야 한다. 다음 아래와 같은 지령으로 그 서고를 푼다.

```
$ tar xzvf /tmp/netscape-i686-pc-linux-gnu-sea_tar.gz
```

이 지령은netscape-installer등록부에 README파일과 netscape-installer스크립트를 비롯하여 여러개의 파일을 작성한다. cd /tmp/netscape-installer 지령으로 이 등록부를 조종하며 X Windows말단으로부터 설치스크립트를 실행하여 설치를 진행한다.

```
$ ./netscape-installer
```

이로부터 Netscape와 함께 설치하려는 프로그람묶음을 선택할수 있는 여러개의 차림표들을 고찰하게 된다.

주의

6.0이상의 Netscape판본을 적재하면 설치순서는 여기에서 본것과는 다르다. 이 경우 **README**파일이나 **INSTALL**파일에 대한 설치파일을 검색하고 **more**지령을 리용하여 파일에 있는 지령들을 조사할수 있다. 의문스러운 점이 있으면 설치과정에 제기되는 모든 문제들에 대한 기정선택항목을 받아 들여야 한다. 이것은 대부분의 구성에서 정교한 설치를 담보하도록 설계되였다.

Netscape 응용프로그람

Netscape 6프로그람묶음에는 다음과 같은 5가지 기본응용프로그람들이 있다.

- 조종기, Web열람기
- 우편 즉 전자우편과 새로운 읽기도구, 통보구성기
- 즉시 Messenger 즉 America Online통보편의프로그람
- 구성기 즉 Web페지를 작성하는 응용프로그람
- 사용자와 사용자가 통신하는 임의의 사용자에 대한 개인정보를 기억하는데 리용되는 주소Book

이 장에서는 Web의 열람과 기초적인 전자우편구성을 고찰한다. 제24장 《전자우편의 읽기》에서는 Netscape전자우편편의프로그람의 리용을 고찰한다.

WWW의 호출

Netscape 6으로 Web의 호출은 추가적인 정보를 요구하지 않는다. 간단히 인터네트에 접속하면 된다(앞의 장에서 설명한것처럼). Netscape 6이 기정 /usr/local/netscape 등록부에 설치되였다고 보면 다음의 지령이 나타난다.

$ /usr/local/netscape/netscape

창문은 그림 23-6과 같이 나타난다.

그러므로 Netscape는 Windows 혹은 Macintosh콤퓨터에서와 거의 같은 방법으로 동작한다.

그림 23-6. 초기Netscape 6창문

주의

만일 Web열람기가 요구되면 http://home.netscape.com으로부터 Netscape Navigator 4.76프로그람묶음을 내리적재할수 있다. 그러나 이 프로그람묶음은 이 책과 함께 제공된 Red Hat Linux 7.1의 Publisher's Edition에 없다.

현재 창문의 맨 아래 왼쪽구석에 있는 련관된 아이콘을 찰각하여 Netscape의 다섯가지 구성요소가운데서 임의의것을 호출할수 있다. 그것들은 Navigator(Web열람기), Mail, Instant Messenger, Composer, Adress Book순서로 왼쪽으로부터 오른쪽으로 놓여 있다

Netscape 의 측면띠

Netscape 6에서 새로운 주요기능은 측면띠인데 이것은 지정된 축소열람기와 기능이 같다. 기정측면띠들은 아주 광범히 리용되는데 이것들은 소식란표제를 통하여 탐색할수 있다.

각이한 측면띠가 요구되면 http://search.netscape.com/mysidebar.tmpl의 My Sidebar Directoty페지를 조종해야 한다. 그러면 열람기의 이 페지로부터 각이한 측면띠들을 활용할수 있거나 그룹이나 조직에 대한 전용측면띠를 만드는 더 상세한 정보를 얻을수 있다.

Search측면띠 그림 23-7에서 보여 준 Netscape Search 측면띠는 현재 기본열람기 창문에 있는것을 블로크화하지 않고도 표준인터네트탐색엔진을 호출할수 있게 한다. Netscape 6에서 리용되는 기정탐색엔진은 Lycos이다. 즉 Preferences창문에서 이것을 변경시킬수 있다. Edit -> Preferences를 선택한다. 열려 진 Preferences창문에서 Navigator -> Internet Search를 선택한다. 그다음 제시된 여러개의 각이한 탐색엔진에서 선택할수 있다.

그림 23-7. Netscape 6 의 Search측면띠

그림 23-7에서 보여 준 실례는 《Grateful Dead Bears》에 대한 탐색을 설명한것이

다. Search Results는 탐색하여 얻어 진 련관항목들을 제시한다. 련관항목들을 찰칵하면 Netscape는 직접 련관된 Web페지를 조종한다.

What's Related측면띠 What's Related측면띠는 탐색과정에 더 많은 정보를 볼수 있게 해준다. 실례로 그림 23-8에서 보여 준 Momma Bears' Bears Web싸이트와 련관된 싸이트들을 보려면 이 측면띠는 적용할수 있는 부류에 따르는 탐색들에 대한 몇가지 항목들을 제공한다. 실례로 Momma Bears' Bears 는 수집능력이 있는 처리기이다. Recreation:...:Dealers를 찰칵하여 이런 형태의 처리기들을 조사할수 있다.

Buddy List 측면띠 AOL Instant Messenger(IM)봉사를 리용하면 AOL IM사용자 즉 동료와 통신할수 있는데 그때 실시간적으로 직결상태에 있다. Buddy List측면띠에서 동료들의 목록을 설정할수 있다. 그전에 Netscape 6에서 Instant Messaging을 리용하지 않았으면 이 측면띠는 설정을 진행하는 조수를 가지고 그뒤의 buddy목록을 관리한다.

그림 23-8. Netscape 6의 What's Related측면띠

Stocks 측면띠 기정 Stocks측면띠에는 U.S.Dow Jones, NASDAQ, S&P 500, AOL에 대한 최근가격들이 들어 있다. 이 측면띠에 stocks를 추가하려면 Edit를 찰칵하고 나타나는 지령들에 응답해야 한다.

News 측면띠 기정News측면띠는 U.S.News, Sports, Politics에서 가장 최근의 표제들을 포함한다. 완전한 내용과 결합하려면 해당한 표제를 찰칵해야 한다.

Today's Tips측면띠 이 측면띠는 추가적으로 이미 구성된 측면띠를 비롯하여 Netscape의 Tips와 Tricks에 대한 결합을 제공한다. 실례로 그림 23-9에서 보여 준 측면띠는 추가적인 측면띠와 결합한다. 이 측면띠에서 Health News나 Home Improvement를 찰칵하면 Netscape는 열람기에 새 측면띠로서 이 항목들을 추가하게 된다.

그림 23-9. Netscape 6의 Today's Tips측면띠

Web페지열기

측면띠특성을 리용하는것외에도 다른 방법으로도 Web페지를 열수 있다. 첫번째 방법은 그림 23-10에서 보여 준것처럼 기본창문의 탐색마당에 직접 요구되는 페지의 URL을 입력하는것이다. Enter를 누르면 Netscape 6은 입력한 URL을 적재한다.

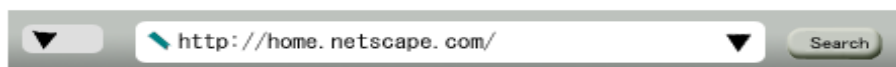



그림 23-10. Search마당

새로운 페지를 여는 다른 방법은 그림 23-11에서 보여 준것처럼 파일차림표로부터 Open Web Location을 선택하는것이다. 이것은 그림 23-12에서와 같은 Open Web Location 대화창문을 펼친다. 열려고 하는 URL을 입력하고 Web페지를 펼치려고 하는 창문을 선택한다.

그림 23-11. 파일차림표

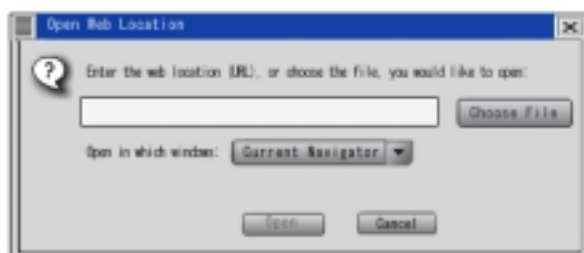



그림 23-12. Open Web Location 대화창문

앞페지로 돌아가기

앞서 본 페지에로 돌아가기는 Netscape도구띠의 Back단추를 눌러서 간단히 진행할 수 있다. 도구띠는 그림 23-13에서 보여 주었다. Back단추는 왼쪽의 첫번째 단추인 왼쪽 방향화살표단추이다.

그림 23-13. Navigator도구띠

그림 23-14에서 보여 준 Go 차림표에서 Back를 선택하여 앞페지에로 돌아 갈수도 있다. Go 차림표는 지금까지 본 문서의 리력을 보여 주며 Go 차림표로부터 그것을 선택해서 임의의 현시된 문서에로 넘어 갈수 있다.

그림 23-14. Go 차림표

페지인쇄

제15장 《체계구성을 위한 Linuxconf와 기타 도구들의 리용》과 제18장 《주변장치의 리용》에서 본대로 인쇄기를 이미 설치하여 구성했다면 Netscape 6으로부터 Web페지들을 인쇄할수 있다.

현시된 페지를 인쇄하려면 파일차림표로부터 Print를 선택하거나 간단히 열람기의 오른쪽 웃구석에 있는 도구띠에의 Print단추을 찰칵할수 있다. 그러면 그림 23-15와 같은 대화칸이 나타난다. 파일을 인쇄하거나 구성된 인쇄기렬을 인쇄하려면 이 대화칸을 리용할수 있다.

그림 23-15. Print대화칸

파일로 인쇄 파일로 인쇄하기는 인쇄되고 있는 페지의 PostScript영상을 만들기 위해 Netscape를 리용한다. 결과파일은 GhostScript와 같은 PostScript보기도구나 PostSCript파일로 조종할수 있는 쏘프트웨어(문서편집기, 도형도구, 기타 응용프로그람들을 비롯한)로 볼수 있으며 후에 인쇄기에로 직접 보낼수도 있다.

파일로 인쇄하기 위해 그림 23-15에서 보여 준 Print 대화칸으로 넘어 가 보자. 창문의 맨 우에 놓인 Print To항목에서 파일을 선택한다. 이것은 File본문칸을 능동상태로 만든다. 그때 그림 23-16에서 보여 준것과 같이 결과파일을 만드는 Netscape에 경로와 파일이름을 입력한다.

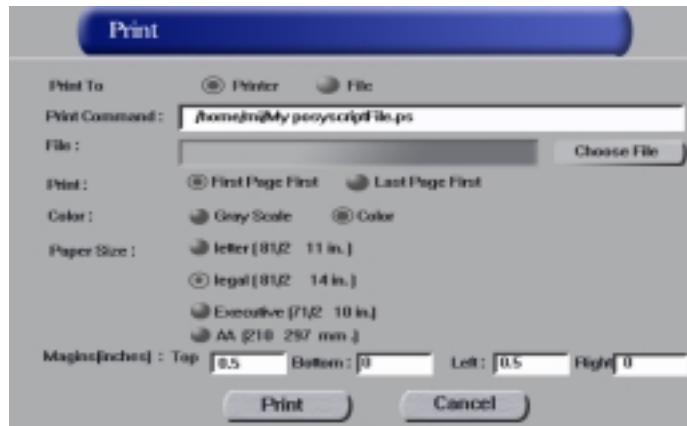



그림 23-16. 파일로 인쇄

인쇄대기렬로 인쇄 인쇄대기렬로 인쇄하기 위해 그림 23-15에서 보여 준 Print대화칸으로 넘어 가 보자. 여기서 Print Command본문칸에 인쇄기에 대한 완전한 인쇄지령을 입력한다. 실례로 laserjet51라는 인쇄대기렬을 가진 인쇄기로 인쇄하려면 인쇄지령 **lpr –Plaserjet51**를 입력해야 한다. 경로를 거쳐서 lpr지령의 완전한 경로를 정의해야 한

다. 이 경우에 인쇄지령은 /usr/bin/lpr –Plaserjet51로 된다. lpr지령은 제18장에서 구체적으로 고찰한다.

주의	일반적으로 Linux에서 lpr지령들은 –P와 및 항목 laserjet51로 된 선택항목사이에 공백을 요구한다. –P선택항목을 가진 lpr지령이 선택항목과 그 항목사이에 공백이 없이 드문히 리용되는 경우도 있다.

First Page First 혹은 Last Page First인쇄상태는 페지들이 인쇄기에서 나오는 방식에 따른다. 일반적으로 아래로 나오게 하려면 First Page First를 선택하고 우로 나오게 하려면 Last Page First를 선택해야 한다.

GrayScale이나 Color항목을 정확히 선택하는것이 중요하다. 만일 흑색인쇄기를 가지고 있다면 GrayScale을 선택해야 한다. 일부 흑색인쇄기들은 Color가 선택되면 리용할수 없으며 읽을수 없게 만들어 놓는다. 만일 색인쇄기를 가지고 있다면 인쇄하려는 방식에 따라서 임의의 방식을 선택할수 있다.

인쇄기에 맞는 정확한 종이크기를 선택해야 한다. 마지막으로 여백들을 선택하고 Print단추를 찰칵한다.

도움말리용

Netscape 6은 쏘프트웨어에 대한 완전한 도움말을 갖추고 있다. 그림 23-17에서 보여 준 Help차림표를 보려면 Help⇨Help Contents를 선택해야 한다.

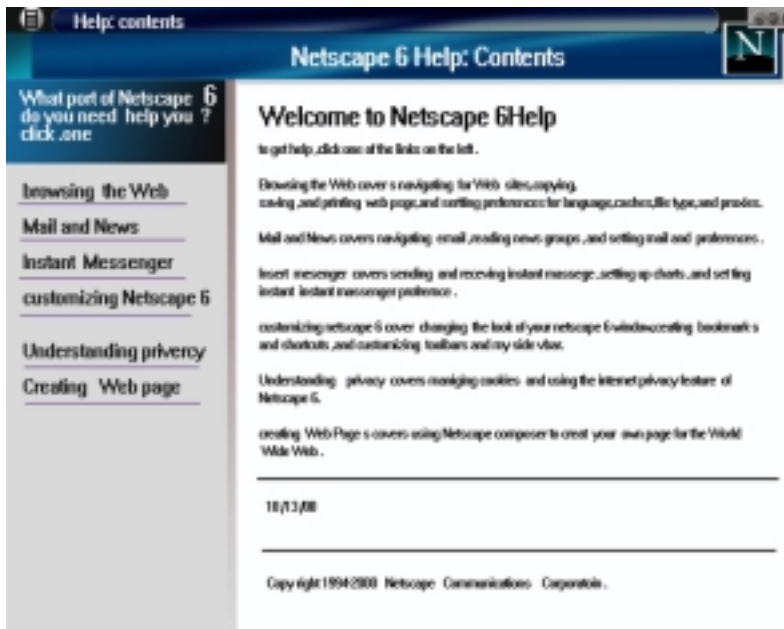



그림 23-17. Help차림표

Netscape 6의 도움말은 6개의 기본항목으로 나누어 볼수 있다.

- 《Web의 열람》에서는 인터네트를 통한 조종, Web파일의 조종, 관련된 언어

번역봉사들의 리용 및 접속에 대한 기명투표 및 기밀보호의 구성에 대한 기초적인 개념을 설명한다.

- 《우편과 새소식》에서는 우편과 새소식봉사기들의 설정, 주소책목록들의 조직, 전자우편과 새소식통보들의 송수신에 대한 기초적인 개념을 설명한다.
- 《지급특별우편통신프로그람》은 Netscape를 리용하여 AOL Instant Messenger응용프로그람의 각이한 특성들을 설정하고 리용하도록 방조한다.
- 《Netscape 6의 전문화》는 책표식들을 포함하여 열람기의 표현을 구성하도록 방조한다. 이 열람기는 차림표항목에 따르는 Web싸이트선택을 조종하게 한다.
- 《기밀의 리해》는 인터네트에 대한 기밀보호를 많이 소개하는데 여기서는 끊임없이 가입하는 임의의 사용자들에게 암호를 요구하면서 관리하는 방법을 설명한다.
- 《Web페지의 작성》은 HTML코드를 배우지 않고도 Web페지를 작성하도록 방조하는 Netscape Composer에 대한 기초적인 개념을 설명한다.

그림 23-17에서와 같이 Netscape도움말의 6개 부류들은 창문의 왼쪽 부분에 조직되여 있다. 이러한 부류중 한개를 선택하면 보조항목설정을 가진 또 다른 창문이 열린다.

Netscape우편구성

Navigator Web 열람기외에도 Netscape 6은 Netscape전자우편이라는 통합된 우편프로그람묶음이 있는데 그것은 POP나 IMAP봉사기에 저축된 우편을 읽기 위한것이다.

Netscap전자우편을 리용하려면 우편을 찾는 방법과 전송하는 방법을 알기 위해 그것을 구성해야 한다. 그보다 먼저 POP와 IMAP봉사기들이 실지 어떻게 국부적이며 원격적인 우편함들로 되는가를 알아야 한다.

국부적이며 원격적인 우편함

국부적인 우편함과 원격적인 우편함의 차이는 포착하기 어렵지만 이 차이는 Linux와 Unix세계에서 각이한 형태의 우편쏘프트웨어를 리해하는데서 기초로 된다.

대부분의 Linux/Unix망에서 사용자들의 우편은 BerKeley Mail Folders(후에 이 형태의 폴더는 버클리에 있는 캘리포니아종합대학에서 발명하였다.)로 알려 진 형식으로 국부우편함에 기억된다. 우편을 주고받으면서 경로에 응답할수 있는 Sendmail과 같은 Linux/Unix우편체계들은 자동적으로 이 표준형식으로 사용자들의 개별적인 우편함에 새로운 통보들을 보관한다. 그다음 사용자들은 우편을 읽고 새 통보문들을 구성하고 그들의 우편함들을 조직하는 표준Unix방식의 많은 우편응용프로그람들을 리용할수 있다.

이 모든것은 Linux체계에서도 리용할수 있다. 모든 일반Unix우편장치들은 Linux에서도 리용할수 있으며 후에 이 책에서 소규모망에 대해 론의할 때 공통적인 해결책으로서 Sendmail과 Unix mail의뢰기를 가진 Berkeley Mail Folder는 전자우편능력들을 가진 사용자들을 제공한다.

그러나 홈사용자에 대한 우편은 일반적으로 LSP가 관리하는 원격봉사기의 우편함에 놓인다. 그것들이 국부적인 BerKeley Mail Folder들인것처럼 이 원격폴더들을 호출하는 쏘프트웨들은 없다. 대신에 POP3이나 IMAP4규약들중 임의의것을 처리하는 우편쏘프트웨어를 리용할수 있다. 이 규약들은 원격우편함을 호출하는 각이한 방법을 제공한다.

주의 POP는 Post Office Protocol의 략자이다. POP3은 정확히 두번째 POP표준이다. 즉 POP2 는1980년대에 일반화되였다. IMAP는 Internet Message Access Protocol의 략자이다. IMAP4는 1995년에 처음으로 공개되였다.

POP3우편봉사기

ISP의 우편봉사기상에 놓여 있는 우편을 호출하는 대부분 Windows나 Macintosh방식의 우편쏘프트웨어는 POP3규약을 리용한다.

POP3모형에서 사용자들은 인터네트에 접속하여 자기들의 우편함으로부터 새로운 우편을 내리적재하고 원격봉사기상에 있는 우편함으로부터 선택적으로 지우거나 복사할수 있다. 이러한 효과는 비직결상태로 읽을수 있는 국부적인 우편함을 작성한다. 국부적인 우편함은 봉사기상에 있는 초기우편함의 내용들을 완전히 복사할수 있다. 사용자들은 통보들, 구성응답들, 새로운 통보들을 읽을수 있으며 비직결상태에서 자기들의 우편을 조직한 다음 자기들이 구성한 통보들을 보내기 위해 재접속하고 새로운 우편을 검색할수 있다.

이것은 간단한 규약이며 모든 사용자들에 대해서 공통적이다. 이것은 직결시간이 비용이 많이 들거나 많은 량의 유용한 대역범위가 제한될 때 해결책으로 된다. 비직결작업상태에 대해서 설계되였기때문에 POP3은 극한된 망호출로 사용자들에게 편의를 도모해준다. 그러나 규약의 단순성으로 하여 일부 제한성도 가진다.

실례로 두개이상의 원격우편함을 호출하려는 사용자가 있다고 하자. POP3을 리용하는 원격봉사기에서 적재된 모든 통신의 복사를 그대로 놔두는것은 있을수 있는 일이다. 이것은 서로 다른 위치에 있는 두개의 PC들우에서 우편쏘프트웨어를 리용하여 모든 통보문들을 내리적재할수 있게 한다. 그러나 통보들이 다른 위치로부터 응답되였는지 안되였는지 알 방도가 없으며 만일 통보가 두 위치에 적재되기전에 어떤 원인으로 원격봉사기에서 지워 졌다면 한개의 위치는 미완성우편함을 가지게 된다.

각이한 두개의 PC를 한개는 집에, 다른 한개는 휴대용으로 된 PC들을 리용하는 사용자의 계획을 보자. 두개의 PC들은 둘다 POP3우편쏘프트웨어를 리용하는 같은 우편함으로부터 우편을 읽는다. 아침에는 먼저 집에 있는 PC가 날자우편을 내리적재한다. 33개의 잡동사니통보들을 포함하여 50여개의 통보들이 있다고 하자. 사용자는 즉시 잡동사니통보들을 지우고 7개의 통보들에 응답하며 후에 보기 위해 10개의 통보들을 파일로 만들었다고 하자.

오후에 그 사용자는 휴대용콤퓨터를 리용하여 자기우편함을 호출한다. 그것들은 자기들의 우편을 내리적재한다. 현재 통보들의 수는 현재 58개이다. 여기에는 집에 있는 콤퓨터에서 지운 33개의 잡동사니통보들, 응답한 7개의 통보들, 후에 보기로 한 10개의 통보들과 추가적으로 새로운 8개의 통보들이 포함된다.

비정상적인 이 상태는 두개이상의 위치에서 같은 우편으로 작업하는데 불필요한 반복을 만들수도 있다.

IMAP4우편봉사기

이 문제에 대한 해결책은 IMAP4 우편봉사기를 리용하는것이다. IMAP모형에서 우편함들은 원격봉사기에 존재하며 거기에 내용들을 보존한다. 사용자들은 의뢰기응용프로그람을 리용하여 이 우편함들을 열고 호출하지만 국부적인 우편함에 대한 통보들을 복사하

지 않고 국부적인 복사물을 매번 만들지 않고도 원격우편함을 간단히 조종할수 있다.

IMAP는 POP3체계에서 생기는 일부 문제들을 해결하는데 여러가지 방조를 준다. 이것은 다중 위치와 콤퓨터들로부터 한개의 우편함으로 작업하고 그 상태를 보존하는 능력을 가진다. 이것은 우편이 다중국부우편함들에 재배치되지 않고 하나의 원격조종우편함에 보존된다는 사실에 기인된다. 만일 사용자가 어떤 위치에 있는 우편함에서 한개의 통보를 지운다면 다른 사용자로부터 그 우편함을 호출할 때 거기에는 그 통보가 없다.

그러나 원격우편함을 조종하는 자연적인 IMAP4는 직결로 우편을 읽고 응답하며 처리할것을 요구한다. 따라서 IMAP4는 망접속이 항구적이거나 본질상 자유로운 경우에나 적합하다. 실례로 정액제료금인터네트가 값이 비싸고 정액제료금국부전화도 비용이 많이 든다면 POP3과 IMAP4우편접속들도 같다. 그러나 만일 사용자가 인터네트와 전화호출비용(미국을 제외하고)을 분당 지불하면 그때 IMAP4가 POP3보다 훨씬 더 많은 비용이 든다는것을 알수 있다.

또한 그리 많지 못한 ISP들은 IMAP4 봉사기들을 통하여 사용자우편함들에 대한 호출을 진행하게 한다. IMAP4체계들은 기업들에 훨씬 더 일반적이며 작업장과 가정들에서 사용자들이 POP3체계들에서 제기되는 어려움을 겪지 않으면서 자기의 우편함을 호출하도록 한다.

POP3과 IMAP4전자우편봉사기의 기본적인 차이를 리해하였으므로 이제는 전자우편을 위한 Netscape 6을 구성할 준비를 한다.

전자우편을 위한 Netscape의 구성

먼저 Netscape Mail전자우편을 연다. Tasks -> Mail을 선택하거나 왼쪽밑에 있는 Mail아이콘을 찰칵할수도 있다. Netscape전자우편을 여는 첫 시각에 그림 23-18에서 보여 준 Account Wizard가 나타난다. 이 조수를 가지고 4가지 형태의 우편회계 즉 ISP Or Email Provider, Netscape WebMail, AOL Account, Newsgroup Account를 설치할수 있다.

주의 Account Wizard가 나타나지 않으면 이미 우편회계가 설정되였다는것을 의미한다. 그 경우 Netscape Mail창문에서 Edit -> Mail/News Account Settings를 선택할수 있다. Account Settings창문에서 Account Wizard를 호출하기 위하여 New Account를 찰칵한다.

대부분의 사용자들은 거의나 ISP Or Email Provider를 설정한다. 이 단계들은 Newsgroup Account를 설정하는것과 비슷하다. Netscape WebMail이나 AOL account가 있을 때 체계에 대한 전자우편주소가 요구된다. ISP Or Email Provider를 선택하고 계속 하기 위해 Next를 찰칵한다.

다음 Account Wizard화면에 그림 23-19에서와 같이 이름과 전자우편주소를 입력한다. 입력한 이름은 전자우편통보와 함께 보게 된다. 여기서 입력한 전자우편주소는 통보문에 추가된다. 통보문에 Reply를 찰칵한 사용자는 그 전자우편주소에 응답을 보내게 된다.

주의 Newsgroup Account를 설정했으면 붙이려는 이름과 전자우편주소를 입력한다.

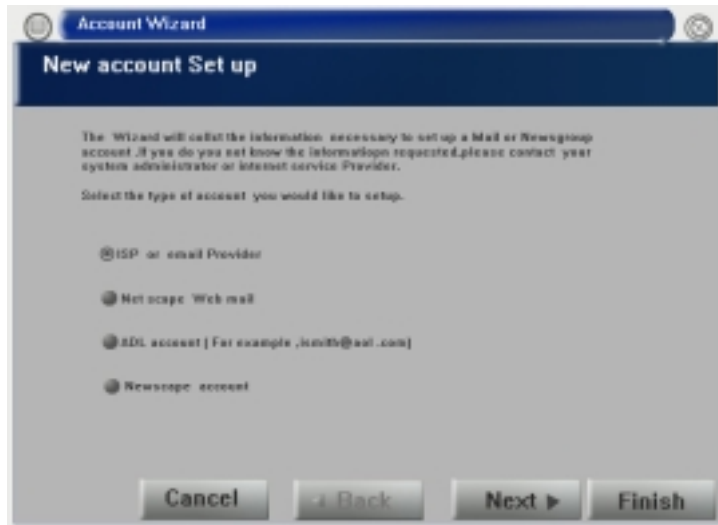



그림 23-18. Netscap Email Account Adress

그림 23-19. 이름과 전자우편주소의 구성

일러두기 광고와 같은 불필요한 전자우편을 피하기 위한 일반방법들중 하나는 전자우편주소에 nospam을 추가하는것이다. 실례로 전자우편주소가 mike@mike.mke라면 mike@nospam.mike.mke 을 입력한다. 통보문으로부터 전자우편주소를 수집하는 콤퓨터는 불필요한 전자우편을 보낼수 없다. 그러나 친구들이나 이웃들은 전자우편주소로부터 nospam을 제거함으로써 전자우편을 전달하도록 한다.

선택된 이름과 전자우편주소를 입력한 다음 Next를 찰칵한다. 그림 23-20에서와 같은 화면에서 전자우편의 받기 및 보내기와 련관된 봉사기들을 구성할수 있다.

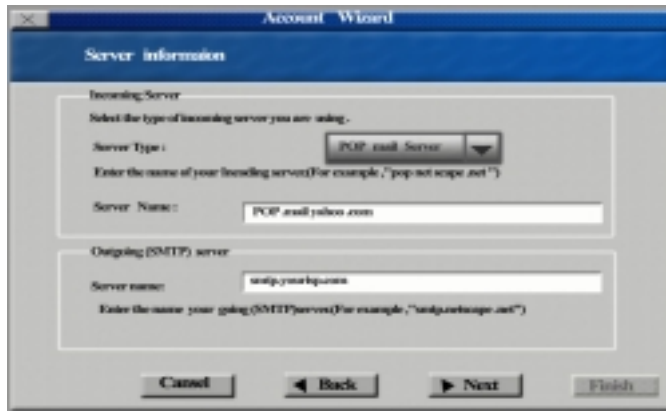



그림 23-20. 우편봉사기의 구성

두가지 형태의 우편수신봉사기 즉 POP3과 IMAP4가 있다. 이미 언급된것처럼 POP3 봉사기들은 ISP, 기타 인터네트우편봉사들과 호상 련관되여 있다. IMAP4는 콤퓨터에 통보문들을 내리적재하기전에 그것을 통하여 탐색하도록 하는 더 능력 있는 우편봉사기이다. 어느 경우에나 관계없이 전자우편주소를 설정하는 회사로부터 우편수신봉사기와 관계되는 이름을 얻는다. 이것은 pop.emailprovider.net 와 같이 나타난다.

우편송신은 ISP의 SMTP봉사기를 통하여 진행되는데 이것은 모든 출력통보문들을 관리한다. ISP로부터 smtp.yourISP.net와 비슷한 봉사기의 이름을 얻어야 한다. 우편수신 및 송신봉사기들의 이름들을 입력한후 계속 하기 위하여 Next를 찰칵한다.

전자우편봉사제공자가 추가한 사용자이름을 입력한다. 많은 경우 전자우편주소에서 기호 @앞에 있는 부분이 사용자이름이다. 사용자이름을 잊어 버렸다면 그에 대하여 전자우편봉사기에 문의할수도 있다. 사용자이름을 입력한 다음 Next를 찰칵한다.

Account Name라는 이름을 가진 다음화면이 제시되며 이 회계에 제시할 이름을 입력할것을 요구한다. 이름을 입력하고 Next를 찰칵한다.

그림 23-21에서 보여 준것과 같이 마지막화면은 이 회계를 만들 때 리용한 자료들을 보여 준다. Back를 찰칵하여 이전 과정으로 돌아 가 잘못된 정보자료를 다시 고칠수도 있다. 자료들이 정확하면 Finish를 찰칵한다.

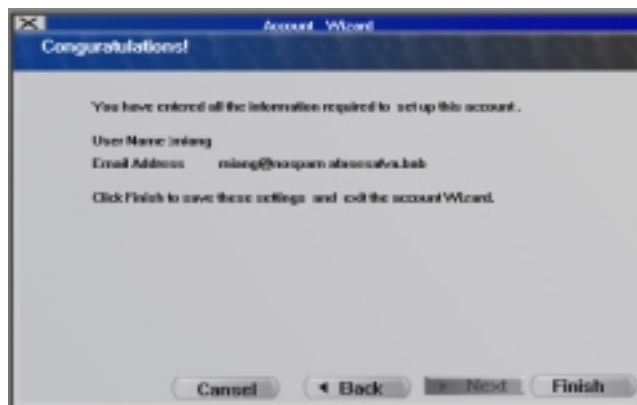



그림 23-21. Account Wizad가 완료된다.

Netscape우편의 재구성

Netscape전자우편에서 설정한 임의의 회계를 재구성하기는 쉽다. 전자우편주소를 설정할 때 다섯개의 각이한 구성항목들을 리용할수 있다. 매 추가적인 전자우편주소에 대해 그 확장주소를 조정하는 3개의 추가적인 구성항목을 취한다. 이 화면들을 호출하기 위하여 Edit -> Mail/News Account Settings를 찰칵한다. 그러면 그림 23-22에서 보는 것과 같은 Account Settings창문이 열린다.

그림 23-22. 회계설정들의 변경

그림 23-22에 보여 준 다섯개의 항목을 보자. 그것들중 세개는 특별한 전자우편주소로 할당된다. 전자우편주소자체는 Account Settings이라고 하는 첫번째 설정이다. 봉사기는 두번째 설정이다. Copies And Folders는 세번째 설정이다. 첫 두개는 전자우편을 설정할 때 구성된다. Copies And Folders설정은 실제적인 우편파일들의 위치들을 포함하며 변경을 거의 요구하지 않는다. 전자우편을 수신하기 위해 POP3이나 IMAP4봉사기들중 어느것을 리용하는가에 따라 정보가 약간 변한다. 기타 조종판들은 필요한 구성항목들을 제공하지만 우편송신(SMTP)봉사기이름을 제외하고는 기초적인 전자우편조작에서 비본질적이다.

Account Settings Panel

전송하는 모든 전자우편에 나타나게 되는 정보(전자우편주소, 이름, 조직과 같은)를 설정하기 위해 Account Settings 조종판을 리용한다. 그림 23-22에 이 조종판을 보여 주었다.

첫번째 항목은 전자우편을 정확히 수신하는 항목이다. Account Name은 전자우편제공자가 할당한 실제적인 전자우편주소와 일치하여야 한다.

다른 중요한 마당은 Your Name과 Email Address인데 Your Name에는 우편수신자들이 알아 볼수 있게 자기이름을 써넣어야 하며 Email Address에는 전자우편의 From행

에 나타날 전자우편주소를 써넣는다.

통보문에 응답할 때 리용하는 우편주소가 전송하는 전자우편의 From행에 있는 주소와 다른 경우에는 Reply-To Address마당에 주소를 써넣어야 한다. Organization마당에 써넣을 정보는 전송하는 모든 전자우편의 머리부에 포함되는 정보이며 Netscape 6을 비롯한 우편열람기들에 현시된다.

Attach This Signature마당에서 전송하려는 모든 전자우편의 끝에 첨부될 표식기호를 포함하는 파일을 정의할수 있다. 이 표식파일은 이름, 주소, 전화번호 또는 기타 색인정보를 포함하는 본문파일이다. 표식을 붙이려면 Attach This Signature검색칸을 설정하여 그것을 활용상태로 만들어야 한다.

Advanced단추는 명명된 회계로 할당된 우편송신(SMTP)봉사기를 변경시킨다.

이 조종판에서 POP3이나 IMAP4봉사기에 대한 내용들과 항목들이 구별된다.

Server Settings Panel

Server Settings조종판은 수신함을 어떻게 찾으며 구성한 우편을 어떻게 전송하는가를 Netscape Mail에 알려 주기 위한것이다. POP3회계가 있다면 그림 23-23과 같이 나타난다.

Server Settings조종판은 POP3이나 IMAP4회계를 리용하는 4개의 기본문자들을 공유한다. 먼저 조종판은 POP3이나 IMAP4의 형태를 가진 봉사기를 구별한다. 다음 Server Name은 ISP에 대한 SMTP봉사기의 제정된 령역이름이다. 세번째 User Name 은 전자우편(ISP에서 리용하는 이름과 일치시킬 필요는 없다)을 리용하는 대상자의 이름이다. 마지막으로 Port는 통보문을 전송하기 위한 접속통로이다. 기정으로 POP3은 110, IMAP4는 143이다.

Server Settings 칸의 기타항목들은 Netscape Mail의 조작에 그리 중요하지 않으며 할당된 검색칸에 표식을 하지 않는 한 비능동상태로 된다. 그 항목들은 POP3이나 IMAP4봉사기를 리용하는데 따른다. 그것들은 기본적으로 자체로 설명된다.즉 항목들의 일부는 전문적인 지식이나 ISP가 만드는 설정에 따른다.

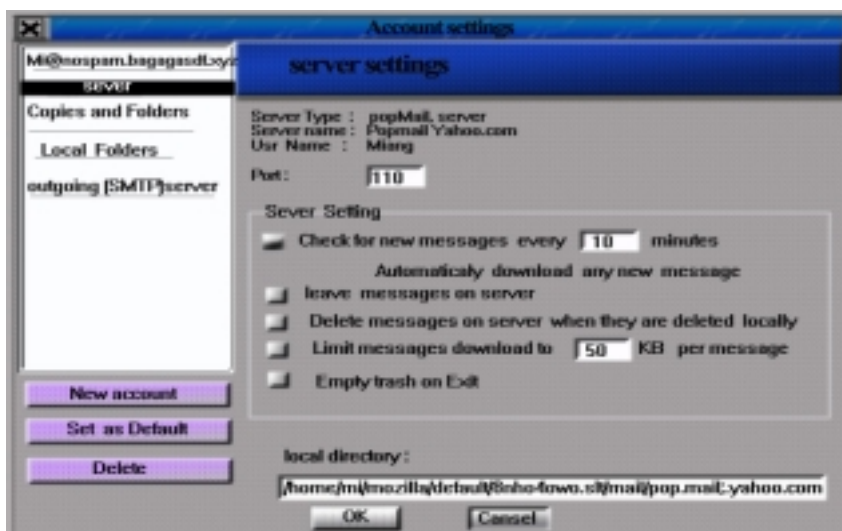



그림 23-23. Server Settings조종판

Lynx설치와 리용

이 장에서 마지막으로 보게 되는 열람기 Lynx는 Unix와 Linux세계에서 리용하는 일반본문열람기이다. Lynx는 이 책과 함께 제공된 많은 Red Hat 배포물들에 포함되여 있다.

Lynx는 다음지령으로 설치되였는가를 검사할수 있다.

```
$ rpm -q lynx
```

Lynx가 설치되지 않았다면lynx-2.8.4-9.i386.rpm을 설치하기 위하여 CD-ROM의 RedHat/RPMS등록부에 있는 rpm을 리용할수 있다. 즉

```
$ rpm -i lynx-2.8.4-9.i386.rpm
```

일단 설치되였으면 /usr/bin/lynx에 Lynx가 설치되게 된다.

새로운 판본의 내리적재

이 책에 서술한 Lynx의 판본은 2.8.4이다. 그것은 현재 쓰는 판본에서 가장 발전된 것이며 Red Hat Linux 7.1에서도 쓸모 있다. 이것을 읽을 때마다 새로운 판본은 http://lynx.browser.org/ 에 있는 Lynx홈페지로부터 리용할수 있어야 한다.

Netscape 6을 적재할 때 리용된 tar.gz형태로 Lynx를 내리적재할수 있으며 그것은 거의나 비슷한 방식으로 설치될수 있다.

Lynx의 기동

Lynx를 리용하기 위해 조종탁이나 GNOME말단과 같은 기타 창문 혹은 xterm창문에서 그것을 기동해야 한다. 지령대면부에서 그것을 기동할수도 있다. Lynx가 있으면 Web을 열람하는데 X Windows가 없어도 된다.

어떤 인수들이 없이 Lynx를 기동하면 Lynx는 자기의 기정페지를 적재하려고 한다. Red Hat Linaux 7.1과 함께 배포된 두번째 판본인 경우 기정페지는 조작체계에 설치된 HTML문서페지이다. 만일 Lynx를 자체로 콤파일했다면 이 페지는 http://lynx.browser.org/ 로 된다.

Lynx가 기동하면 그림 23-24에서 보여 준것과 같은 화면이 나타난다. 이 그림은 Red Hat Linux 7.1에 설치된 국부방조파일을 보여 준다.

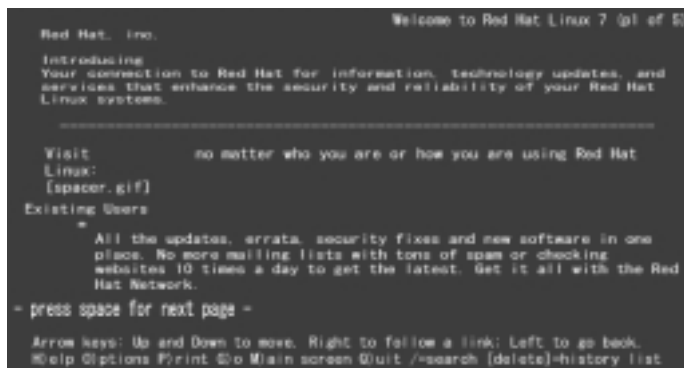



그림 23-24. Red Hat 방조파일을 가진 Lynx

만일 Lynx를 기동할 때 지정한 페지나 파일을 직접 호출하려면 간단하게 다음지령을 리용하여 Lynx에 인수로 페지URL이나 파일이름을 제공해야 한다.

```
$ lynx http://www.yahoo.com/
```

또는 HTML문서를 다음 지령으로 조사해야 한다.

```
$ lynx /tmp/filename
```

Lynx는 자기의 동작을 바꿀수 있는 여러가지 기발들을 가진다. –help인수로 완전한 목록을 볼수 있다. 즉

```
$ lynx –help
```

일반적으로 Lynx로 Web을 열람할 때 이 지령행인수들을 리용하지 않아도 된다.

새로운 페지의 열기

Lynx화면의 아래행에 Web을 열람하는데 리용할수 있는 기본주요지령이 현시된다. Lynx를 실행할 때 새로운 페지를 열기 위해 G건으로 Go지령을 실행한다.

Lynx는 다음과 같은 재촉문에 응답한다.

URL to Open:

인터네트에 이미 접속되였다고 보면 www.mommabears.com과 같은 URL을 입력하고 Enter건을 눌러 페지를 열수 있다.

련결의 잇기

문서에서 련결은 xterm이나 조종탁창문에서 실행될 때 천연색본문에 나타난다. 능동련결은 붉은색으로 나타나고 다른 련결들은 푸른색으로 나타난다. Enter건을 눌러 능동련결을 잇는다. Tab건을 리용하여 다른 련결들을 능동화시킴으로써 련결을 통하여 앞으로 이동할수도 있고 웃방향 및 아래방향건을 리용하여 다른 련결들을 능동화시킴으로써 앞뒤로 이동할수도 있다.

일단 이으려는 련결이 선택되기만 하면 Enter건을 누르거나 오른쪽방향건을 눌러서 련결을 이어 나갈수 있다. 왼쪽방향건을 리용하여 이미 본 페지로 돌아 갈수 있다. 많은 Lynx지령들을 표 23-1에 주었다.

표 23-1. Lynx지령

명 령	설 명
Enter	붉은색으로 나타난 현재 능동련결을 이어 나간다.
Tab	능동련결을 앞으로 이동시킨다.
왼쪽방향건	이미 본 페지로 돌아 간다.
오른쪽방향건	붉은색으로 나타난 현재 능동련결을 잇는다.
웃방향건	능동련결을 뒤로 이동시킨다.
아래방향건	능동련결을 앞으로 이동시킨다.
H	Lynx도움말을 호출한다.
O	Lynx구성선택항목들의 차림표를 펼친다.
P	인쇄선택항목들의 차림표를 펼친다.
G	www.sybex.com과 같이 조사하려는 Web싸이트를 입력할수 있는 선택항목을 연다.

표계속

명 령	설 명
M	현재 구성된 홈페지를 조종한다.
Q	Lynx를 끝낸다.
/abc	abc문자렬순서로 탐색한다.
Delete	이미 본 Web페지들의 리력을 보여 준다.

영상보기

xterm창문에서 Lynx를 실행한다면 gqview와 같은 도형보기도구프로그람을 리용하는 개별적인 창문의 페지에 내장된 영상을 볼수 있게 하기 위하여 그것을 구성할수 있다.

이를 위하여 그것이 영상을 현시하는 gqview를 리용하는가 알기 위해서 Lynx를 구성하여야 한다. 뿌리사용자로서 favorite본문편집기를 리용하는 파일 /etc/lynx.cfg를 편집하여 이 작업을 할수 있다.

이 파일에는 #XLOADIMAGE_COMMAND로 시작하는 두개의 행들이 있다. 한개의 행은 VMS머리부아래에, 다른 하나의 행은 Unix머리부아래에 나타난다. 사용자는 Unix머리부아래에 나타난 행으로 작업하게 된다. 만일 gqview를 리용하려면 이 행을 읽을수 있게 변경해야 한다. 즉

XLOADIMAGE_COMMAND:/usr/bin/gqview %s &

행의 앞부분에 있는 파운드기호(#)릴 삭제할 때 주의를 돌려야 한다. 이것은 Lynx가 적재할 때 그것에 주목을 돌리게 하도록 하기 위해서 행을 설명하지 않는다.

일단 이 작업을 진행했으면 변경된 파일을 보관하고 Lynx를 기동해야 한다. 다음 Images-As-Links방식으로 돌아 가야 한다. 별표(*)지령은 Images-As-Links방식을 고정시킨다. 별표건을 누르면 다음의 통보문이 나타난다.

Links will be included for all images! Reloading...

Lynx는 /etc/lynx.cfg파일에 설치한 프로그람의 영상련결을 연다. 다음의 통보문이 나타나면 어떤 보관된 영상과도 련결할수 없다.

Standard image handling restored! Reloading...

▶미리 알아두기

지금까지 Linux를 리용하여 Web을 직결하고 열람하는 방법을 보았다. 그러므로 기타 주요인터네트기능 즉 전자우편에 대한 완전한 선택항목들을 고찰할수 있게 되였다.

이미 대중적인 우편읽기응용프로그람인 Netscape 6을 보았다. 그러나 Netscape는 선택할수 있는것들중의 하나일뿐이다.

Linux는 상업 및 무료우편의뢰기의 방송범위를 제공한다. 이것들은 강력한 재촉문지향체계로부터 광범한 문자방식우편읽기장치 및 보잘것 없는 문자방식우편읽기도구(잘 알려 진 pine 및 elm프로그람묶음과 같은), 도형우편읽기도구(Xmail과 같은)에 이르는 범위를 포괄하고 있다.

제24장《전자우편의 읽기》에서는 Linux에서 리용할수 있는 우편읽기도구를 넓은 범위에서 고찰한 다음 더 일반적인 리용가능한 우편읽기도구들의 일부를 구체적으로 조사하게 된다.

제24장. 전자우편의 읽기

이 장에서는 Linux에서 전자우편을 읽고 구성하고 전송하는데 리용될수 있는 각이한 수단들과 프로그람들을 고찰한다. 여기에는 비직결우편읽기도구들을 설명하는 Netscape 6우편편의프로그람이 속하는데 이 프로그람은 전자우편통신규약POP3을 리용한다. 비직결우편프로그람의 실례로 Linux본문방식pine전자우편읽기도구를 들수 있는데 이에 대한 프로그람은 전자우편통신규약 IMAP4를 리용한다. POP3과 IMAP4에 대해서는 이미 제23장 《World Wide Web의 리용》에서 보았다.

비직결우편읽기도구

많은 사용자들이 원격우편함을POP3으로 호출하는것은 당연하다. 이것은 밀폐된 CD-ROM에서 리용할수 있는 Netscape 6 을 비롯하여 Linux를 위한 많은 우편응용프로그람들중의 어느 하나를 리용하여 실행할수 있는데 이 Netscape Communicator는 Web싸이트 http://home.netscape.com/ download에서 내리적재할 때 리용할수 있다.

주의 이 장에서 리용된 Netscape응용프로그람은 ftp://ftp.netscape.com/pub/netscape6/english/6.0/unix/linux22/sea/ 로부터 내리적재할수 있다. 이것을 읽을 때 최신판본을 이 싸이트상의 다른 등록부로부터 직접 내리적재할수 있어야 한다.

이 절에서는 오늘날 가장 대중적인 도형우편의뢰기의 하나이며 Windows와 Macintosh로부터 Linux에로 방향전환한 사용자들이 이미 Netscape의 제품에 대해서도 잘 알고 있어야 하기때문에 Netscape 6을 고찰한다. Linux에는 리용할수 있는 많은 POP와 IMAP 우편읽기도구들이 있다. 프로그람묶음을 선택하는 목록은 http://www.linuxapps.com의 Linux Applications Pages에 있는 우편부류들을 검색할수 있다.

주의 우편봉사기가 우편통신규약 IMAP을 리용하면 직결우편읽기도구로서 Netscape를 리용할수 있다.

Netscape 6

원격 POP3 우편함을 읽고 이 우편함을 통보문비직결을 구성하는데 리용하며 그것들을 순서화하기 위해서 Netscape 6을 아주 간단하게 구성할수 있다. Netscpae 6을 실행하려면 xterm창에 다음의 지령이 나타나야 한다.

```
$ /usr/local/netscape/netscape
```

이때 기정위치에 Netscape를 설치했다고 가정한다(제23장에서 Netscape 6를 설치하기에 대한 절을 보시오.).

주의	Red Hat Linux 7.1에 있는 Netscape Messenger 4 .76은 이 장에서 취급하지 않는다. 비록 많은 상세한 차이점들이 있어도 전자우편을 송신하고 수신하기 위한 기초적인 수속들은 변경되지 않는다.

전자우편을 위한 Netscape 6의 구성

일단 Netscape 6을 실행하면 제23장의《Netscape 우편의 구성》절에서 본 단계에 따라 Preferences 대화칸으로부터 우편읽기도구를 구성할수 있다.

Netscape Mail로 우편읽기

일단 우편함을 호출하는 Netscap전자우편이 구성되였으면 통보문의 읽기를 시험해볼수 있다. 그러나 우편함을 호출하기전에 두 작업을 하여야 한다.

1. 인터네트에 접속한다(제22장《Linux를 인터네트에 접속》에서 본 방법들중의 임의의 방법에 따른다.)
2. Netscape 6을 실행한다.

기정으로 Web열람기창문이 열리면 Netscape 6이 실행된다. 우편함창문은 두가지 방법으로 열수 있다. 즉

- 왼쪽 아래구석에 있는 우편아이콘을 찰칵한다. 그것은 봉투기호인데 왼쪽으로부터 두번째이다.

- 그림 24-1에서 보여 준것처럼 Tasks차림표로부터 Mail을 선택한다.

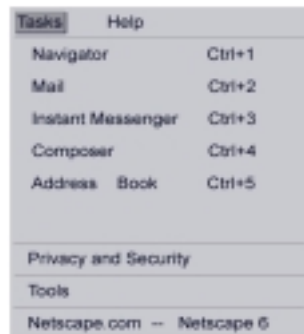



그림 24-1. Tasks차림표

Netscape Mail창문은 그림 24-2와 같다.

Netscape전자우편창문은 4개부분으로 되여 있다. 왼쪽 웃구석의 Mail Folders칸은 전자우편회계로 할당된 여러개의 우편폴더들을 표시한다. 이 폴더들에는 다음과 같은것들이 포함된다.

- Inbox는 우편접수봉사기(POP3,IMAP4)를 통하여 들어 온 우편을 수집한다.
- Drafts는 편집과 배포를 위한 통보문을 보관한다.
- Templates는 전송된 전자우편에 대한 형태와 배경색을 포함한다.
- Sent는 이 회계로부터 전송된 전자우편통보문을 수집한다.
- Trash는 삭제한 통보문을 수집한다.

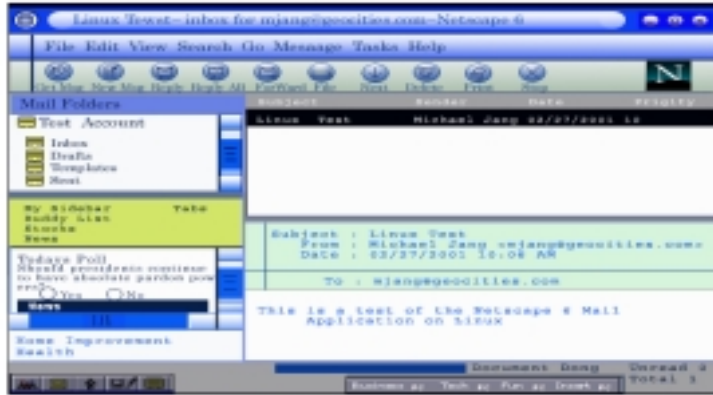



그림 24-2. Netscape Mail창문

오른쪽 웃구석의 List현시판은 선택된 폴더의 통보문목록을 표시한다. 그림 24-2의 Inbox에는 한개의 통보문밖에 없다.

오른쪽 아래의 Message현시판은 List현시판에서 선택된 통보문의 내용을 보여 준다. 그림 24-2의 Message현시판에서 《Linux Test》통보문의 내용을 볼수 있다.

Netscape Mail은 HTML 우편을 지원하기때문에 HTML을 리용하여 형식화된 우편통보문을 Web페지에서처럼 Message칸에 표시할수 있다. 기대한대로 Links가 강조되며 찰칵준비상태로 된다. 한개의 련결항목을 찰칵하면 우편함창문대신에 열람기창문에 요구되는 문서가 열린다.

왼쪽 아래구석의 My Sidebar칸은 Search와 What's Related를 제외하고는 Netscape Navigator의 모든 측면띠들을 포함한다.

Netscape Mail에서 통보문의 구성

통보문을 읽는 과정에 새로운 통보문을 구성할수 있다. 새로운 통보문구성은 매우 간단하다. 즉 New Msg단추를 찰칵하면 된다. 이때 그림 24-3과 같은 통보문구성창문이 나타난다.

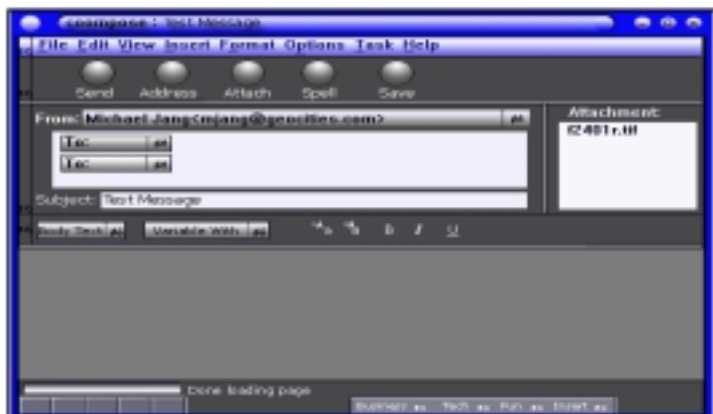
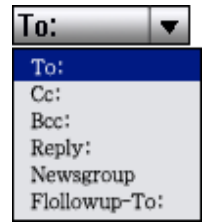



그림 24-3. 통보문구성창문

통보문을 처리하려면 To마당의 옆에 있는 마당에 주소를 써넣어야 한다. 주소를 새롭게 작성한후에 Enter건을 눌러서 다른 사람에게도 보낼수 있게 다음행의 To마당을 비게 한다. 공개사본(Cc마당) 혹은 비공개사본(Bcc마당)과 같은 통보문을 일부 수신자들만이 수신하게 하려면 주소를 변경하기 위하여 To 마당열에 있는 아래방향건을 찰칵해야 한다. 그러면 정확한 선택항목을 정의할수 있는 주소마당차림표가 나타난다.

첨부를 통보문에 추가하기 위하여 도구띠의 Attach단추를 찰칵한다. 그러면 그림 24-4와 같이 홈등록부의 모든 파일목록을 보여 주는 Enter File to Attach창문이 현시된다. 여기서 체계의 임의의 파일을 전송할수 있다.

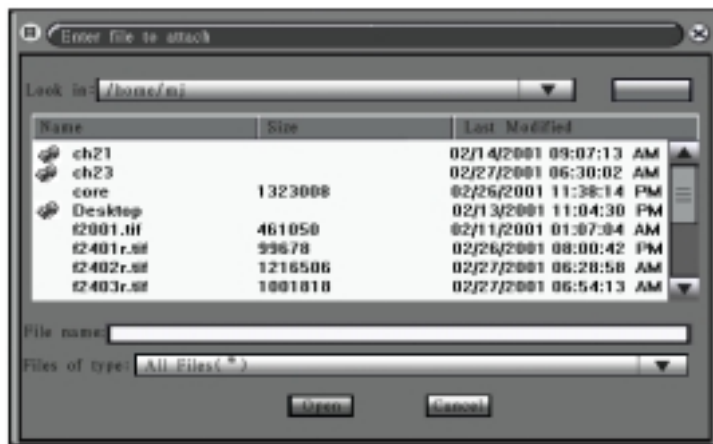



그림 24-4. Enter File To Attach창문

한개 파일을 전송하면 전송된 파일이름(실례에서는 f2401r.tif)은 그림 24-5에서와 같이 통보문의 오른쪽 웃구석의 Attachment칸에 기록된다.

그림 24-5. Attachment칸에 나타난 첨부파일

완성된 통보문은 그림 24-6 에서 보여 준것처럼 Subject마당의 통보문제목과 본문편집칸의 통보문본문으로 구성된다.

그림 24-6. 완성된 통보문

마지막으로 통보문을 전송하기 위하여 Send단추를 찰칵한다(통보문구성이 끝났으면 인터네트에 접속되였다고 본다).

통보문의 배렬 보통 인터네트제공자와 접속한 POP3 우편사용자들은 비직결인 경우에 통보문들을 조립하며 후에 ISP에 접속될 때 배포통보문들을 배렬한다. 이런 조작은 아주 쉽다. 통보문이 구성되면 Windows 도구띠위치에서 Send단추를 찰칵하지 않고 그림 24-7의 File차림표에서 Send Later를 선택할수 있다. 그다음 후에 배포하기 위하여 통보문을 배렬한다.

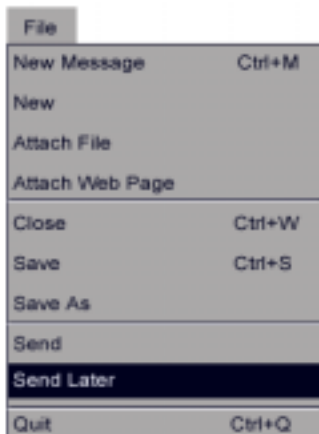



그림 24-7. 통보문구성 File 차림표

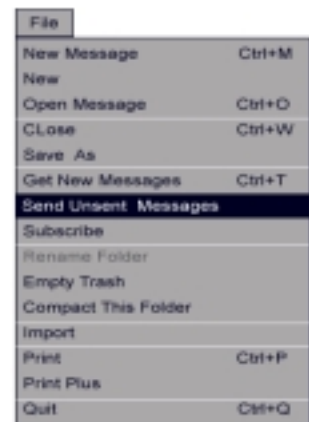



그림 24-8. Netscape 의 Mail's File 차림표

통보문의 구성이 끝났으면 인터네트에 접속하고 그림 24-8의 주요Netscape Mail창문의 File차림표에서 Send Unsent Message를 선택해야 한다.

그러면 전송하지 못한 통보문 즉 배렬된 통보문이 발송된다.

통보문의 응답과 넘기기 새로운 통보문의 구성하기와 통보문에 응답하기 그리고 다른 사람에게 통보문의 넘기기는 밀접히 련관되여 있다.

모든 사람들에게 있어서 가장 일반적인 기능이기때문에 응답을 먼저 보자. 통보문에 응답하려면 우편함에서 통보문를 선택하고 도구띠에 있는 Reply단추를 찰칵해야 한다.

그러면 그림 24-9와 같이 이미 주소화된 통보문의 구성창문이 나타난다. 기정으로 Netscape Mail은 Message칸의 초기통보문내용을 가지고 있다.

마찬가지로 통보문을 넘기려면 통보문를 선택하고 Forward단추를 찰칵해야 한다.

그림 24-9. 통보문에 응답하기

그러면 그림 24-10과 같이 자기의 통보문으로 처리된 선택된 통보문을 가진 주소가 없는 통보문구성창문이 나타난다. Subject행은 전송된 통보문을 나타낸다.

그림 24-10. 한개 통보문의 넘기기

직결우편프로그람

이 절에서는 국부 Unix/Linux 우편함들을 읽기 위하여 설계된 우편프로그람 즉 IMAP4봉사기를 호출할수 있는 우편프로그람에 대하여 고찰한다. 인터네트와 같은 견고한 망에 접속되였다고 보면 Netscape 6 과 같은 비직결쏘프트웨어로부터 이것을 식별할수 있다.

pine 및 elm

Unix나 Linux세계에서 pine과 elm은 가장 일반적인 우편프로그람이다. 이 프로그람들은 조종탁이나 xterm창문에서도 리용되여야 하기때문에 문자방식환경에서 작업하도록 설계되였다. 이것들은 둘다 국부 Berkeley Mail방식봉사기들로부터 호출되도록 설계되였으며 구성할수 있는 많은 선택항목들에 대해서 아주 유연하다.

주의 elm은 pine보다 먼저 나왔다. pine의 사상은 사용자가 조종하기 쉬운 elm의 일부 측면을 받아 들인것이다. pine은 《Pine은 Elm이 아니다》를 의미하는것으로 알려 졌다. pine은 IMAP봉사기들에서 우편을 읽을수 있지만 elm은 읽을수 없다.

elm과 pine들은 다 많은 대학생들이 전자우편을 호출하는 장소인 대학들에서 개발되였다. 이 절에서는 pine에 대해서만 설명한다. elm과 함께 고찰하는 pine은 아주 주목할만하다. 많은 Linux와 Unix사용자들은 pine 대신에 elm을 리용할수 있다고 주장하지만 공교롭게도 elm은 더는 존재하지 않는다. 그 실례로 elm의 마지막계렬이 2.5.5이지만 http://www.math.fu-berlin.de /~guckes/elm의 elm홈페지에서 가장 쓸만하게 유일한것은 문서화된 판본 2.5.3변경일뿐이다.

바로 이것이 pine을 선택하게 된 동기이다. elm과 pine은 유력한 프로그람들이지만 결코 다른것에 비해 볼때 그렇지는 않다. 대다수의 사용자들은 자기의 우편프로그람을 리용한다. 그것은 오늘날 PC가 처리집중도형응용프로그람실행능력보다 더 좋기때문이다. 여기서는 사용자들에게 이러한 전통적인 Unix/Linux 우편프로그람들에 대한 인식을 주기 위하여 pine을 선택하였다. pine을 선택하면 Linux를 지향하는 사람들에게 더 친근감을 줄수 있으며 어떤 문서는 pine 문서로 갱신하는것이 더 쉽다. 체계관리기로 작업하는 모든 대학들에서 전자우편응용프로그람으로서 pine을 리용한것이 아주 좋았다.

pine의 리용

얼핏 보기에는 pine이 완성된것 같지만 실지는 그렇지 않다. 전자우편을 송신하거나 수신하려고 하는 사용자들은 pine을 아주 쉽게 리용할수 있다.

대부분의 Linux체계들은 설치중에 망응용프로그람이 선택되면 설치된 pine을 호출하며 pine이 아직 설치되지 않았으면 http://www.washington.edu/pine/ 의 홈페지에서 pine을 호출한다. Red Hat Linux 7.1에 들어 있는 pine의 판본은 4.33-8이다.

프로그람을 실행하기 위하여 다음의 지령을 간단히 입력한다.

```
$ pine
```

거의 모든 Linux체계에서 기정구성은 원활하게 작업해야 한다.

첫 기동시에 pine은 그림 24-11에서 보여 준것처럼 자기의 기정기본차림표를 현시한다.

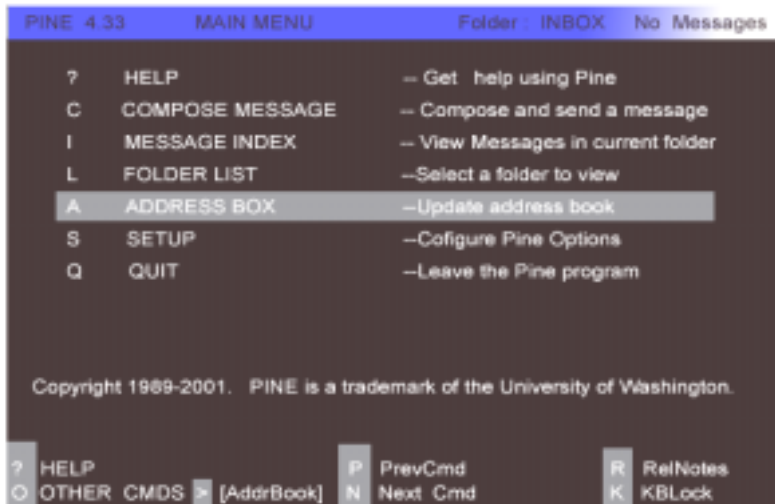



그림 24-11. pine기본차림표

pine화면은 4개의 부분으로 구성되여 있다.

- 웃행은 기동하고 있는 pine의 판본번호, 작업등록부(첫 기동시의 INBOX)의 이름과 현재 존재하는 통보문을 표시한다.
- 밑의 두 행은 현재상태에서 호출할수 있는 일반적인 지령들을 표시한다.
- 나머지화면부분은 기본작업구역을 나타낸다.

기본차림표에는 4개의 기본선택항목이 있다.

C(COMPOSE MESSAGE)는 새로운 통보문을 구성하고 전송한다. 즉 기본폴더의 내용을 보기 위하여 리용된다.

I(MESSAGE INDEX)는 현재 등록부의 내용을 보여 준다.

L(FOLDER LIST)은 리용가능한 등록부들의 목록을 보여 준다.

A(ADDRESS BOOK)는 주소기록부를 호출한다.

이 기능들가운데서 두가지 즉 Folder List와 Compose Message만을 고찰한다. 모든 항목들은 직결방조기능으로 문서화되여 있으므로 기본차림표에서 HELP지령을 리용할수 있다.

Folder List의 열기와 작업 폴더목록을 열기 위하여 아래 혹은 웃방향건을 리용하여 FOLDER LIST를 선택하고 Enter건을 누른다. 또는 L을 선택한다. 그러면 그림 24-12와 같이 리용할수 있는 폴더목록이 현시된다.

이 폴더목록은 현재 리용할수 있는 폴더 즉 이 경우에는 INBOX와 sent-mail, saved-messages들을 보여 준다.

통보문색인의 열기와 작업 폴더목록으로부터 Enter건을 눌러 필요한 폴더를 선택한다. 그림 24-13은 선택된 폴더의 통보문목록을 보여 준다. 화면의 제일 웃행은 현재 선택된 통보문, 폴더에 있는 통보문번호, 통보문의 상태(새로운 통보문을 위한 NEW, 아직 읽어 보지 않은 통보문, 응답통보문을 위한 ANS과 같은)를 보여 준다.

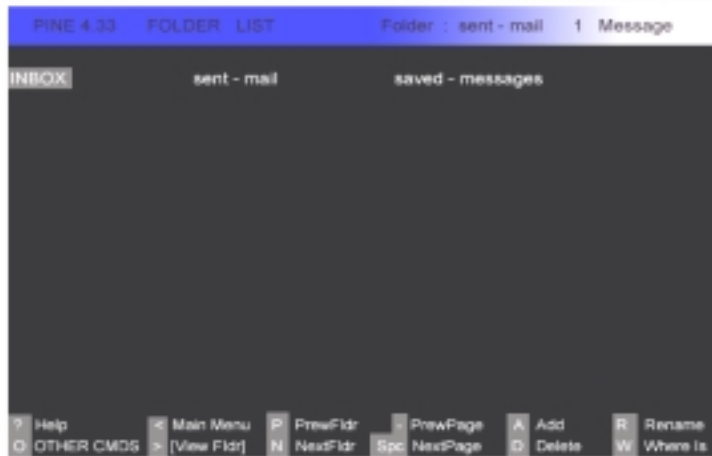



그림 24-12. 현재 pine폴더목록

inbox의 매 통보문은 왼쪽에서 오른쪽으로 가면서 통보문상태(새로운 통보문은 N, 응답통보문은 A, 삭제된 통보문은 D) 를 한 행으로 보여 준다. 통보문에는 통보문번호와 통보문을 받은 날자, 발송자, 통보문의 바이트크기, 통보문의 제목 등이 포함된다.

필요한 통보문을 방향건을 눌러 선택한다. 이때 이전 통보문에로 이동하려면 P건을 누르고 다음통보문에로 이동하려면 N건을 누른다.

현재 선택된 통보문를 삭제하려면 D를 입력해야 한다. 그러면 삭제하려는 통보문을 표시하고 삭제된다는것을 보여 주기 위해 통보문의 상태지시기가 변경된다. 그러나 pine을 그만둘 때까지 통보문은 삭제되지 않는다. 삭제하였던 통보문을 다시 복귀하려면 그 통보문을 선택하고 U를 입력해야 한다.

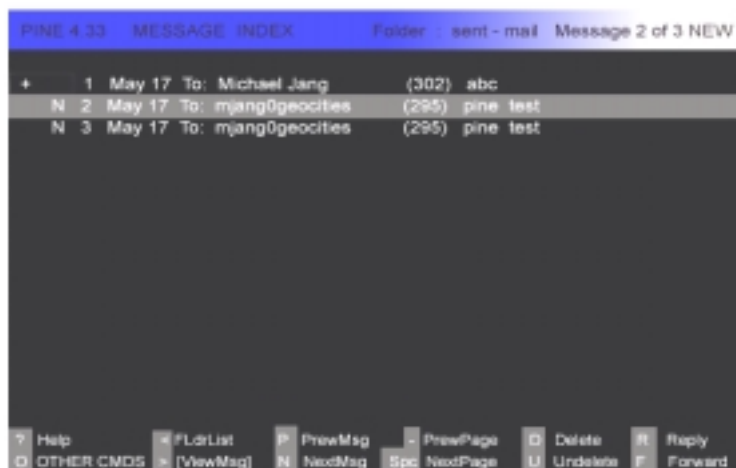



그림 24-13. 폴더에 있는 통보문목록

폴더색인이 한 화면보다 더 길면 공백건과 이음표건을 리용하여 인차 페지절환을 할 수 있다. 해당한 통보문에로 직접 넘어 가자면 지령재촉상태일 때 J지령과 통보문번호를 리용해야 한다.

통보문을 선택하고 Enter건 혹은 통보문보기지령 V를 입력하여 폴더안의 통보문을 읽어 볼수 있다. 통보문은 그림 24-14에서와 같이 나타난다.

통보문을 읽는동안 폴더에서 통보문이행지령 P와 N을 입력하고 D와 U를 입력하여 통보문을 삭제하거나 복귀할수 있다.

통보문이 한 화면보다 더 길면 공백건과 이음표건을 리용하여 화면을 아래 또는 우로 옮길수 있다. 혹은 한행씩 우로 또는 아래로 옮기기 위하여 웃방향 및 아래방향건을 리용할수 있다.

폴더색인에서 기본차림표로 돌아 가거나 통보문보기에서 기본차림표로 돌아 가려면 간단히 M를 입력해야 한다.

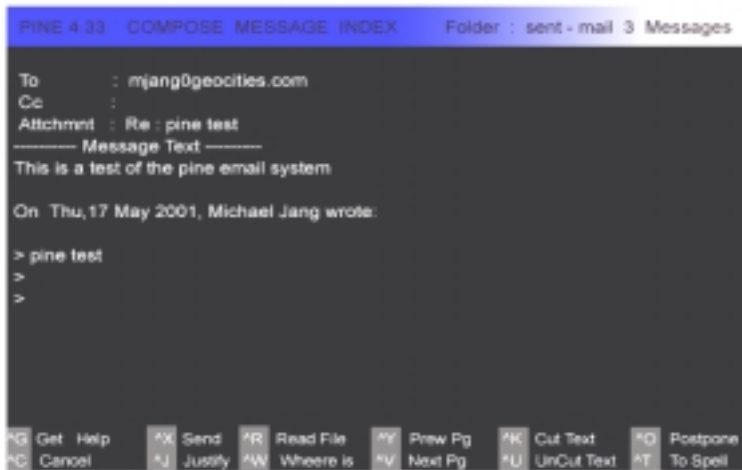



그림 24-14. 통보문보기

통보문구성 기본차림표에서 또는 폴더색인이나 통보문을 읽는동안 통보문을 구성할 수 있다. 이를 위해서 C를 입력한다(기본차림표에서 COMPOSE MESSAGE선택항목을 선택하고 Enter건을 누른다.). 그러면 그림 24-15와 같은 통보문형식이 나타난다.

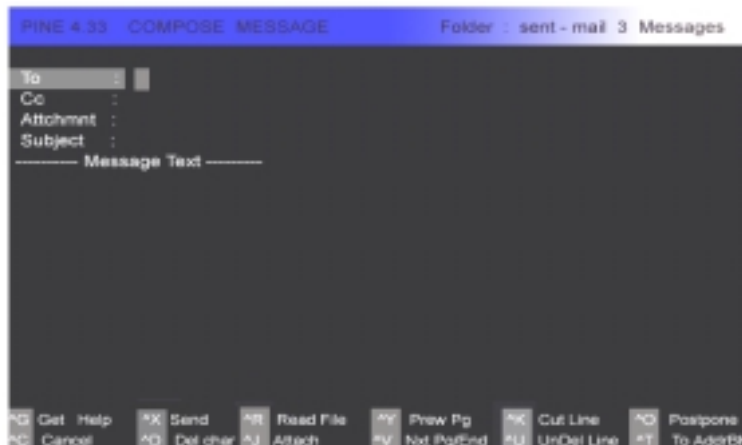



그림 24-15. 통보문구성화면

이 화면은 두개의 부분 즉 Message Text선을 기준으로 우와 아래부분으로 갈라 본다. 웃부분에서는 수신자와 제목, 다른 내용을 정의할수 있으며 아래부분에서는 통보문을 입력할수 있다. 또한 웃방향 및 아래방향건을 리용하여 항목과 마당사이로 이동할수 있다.

To와 Cc마당은 반점으로 구분된 전자우편주소들을 가진다. 유표가 그 마당을 지나가면 그림 24-16과 같이 그 마당의 주소들이 강조되여 현시된다.

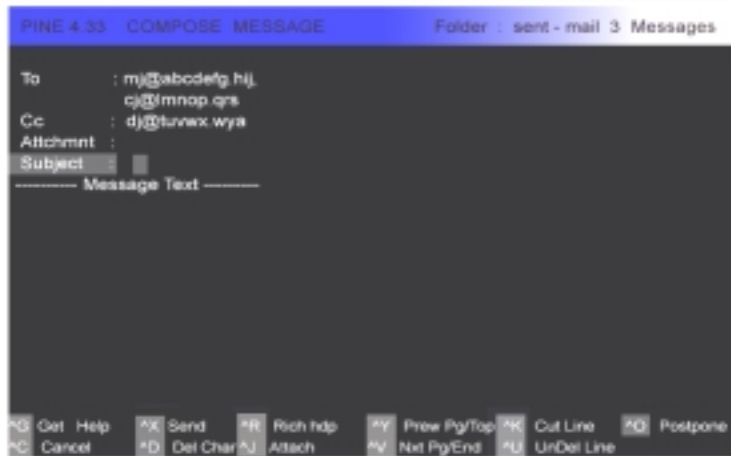



그림 24-16. 다중통보문수신자

비공개사본을 발송하려면 유표가 머리부에 있을 때 Ctrl+R를 눌러서 리용하고 있는 통보문머리부마당을 확장한다(통보문본문안에서는 이 조작을 쓸수 없다.). 이렇게 되면 그림 24-17과 같이 원만한 머리부특징을 호출한다.

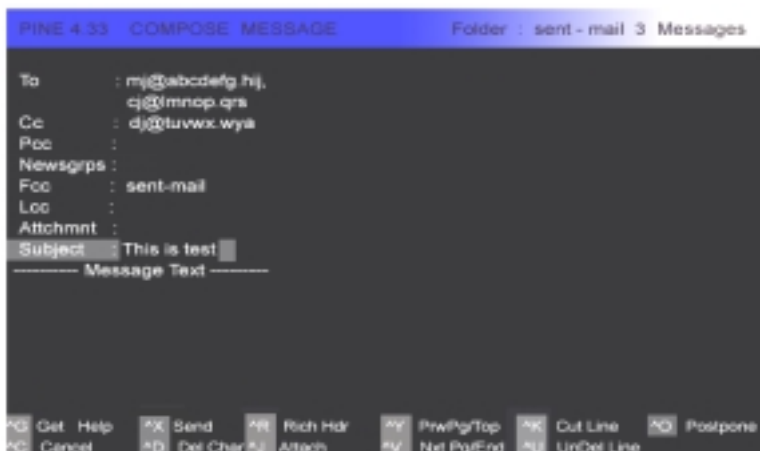



그림 24-17. 원만한 머리부

여기서 전자우편통보문의 비공개사본을 발송하기 위하여 Bcc마당을 찾는다. 다른 마당들은 송신하는 통보문의 복사를 기억시키거나 새소식묶음에 우편물통보문을 복사하도

록 위치를 지정하는것과 같은 많은 기능을 제공한다.

Message Text선 아래에서는 통보문의 본문을 입력할수 있다. 그림 24-18은 완성된 통보문을 보여 준다.

작업을 끝내면 두가지중 하나를 선택한다. 즉 통보문을 발송하거나 그 통보문을 무효시킬수 있다. 통보문을 발송하려면 Ctrl+X건을 누른다. 그러면 통보문을 발송하겠는가를 확인하는 재촉문이 제시되며 거기에서 y 또는 n 을 눌러 긍정 혹은 부정응답을 한다. 그 통보문를 취소하려면 Ctrl+C를 누르고 제시되는 취소확인문에 긍정 또는 부정으로 응답한다. 통보문이 무효되였으면 다시 그 내용을 회복할수 없다.

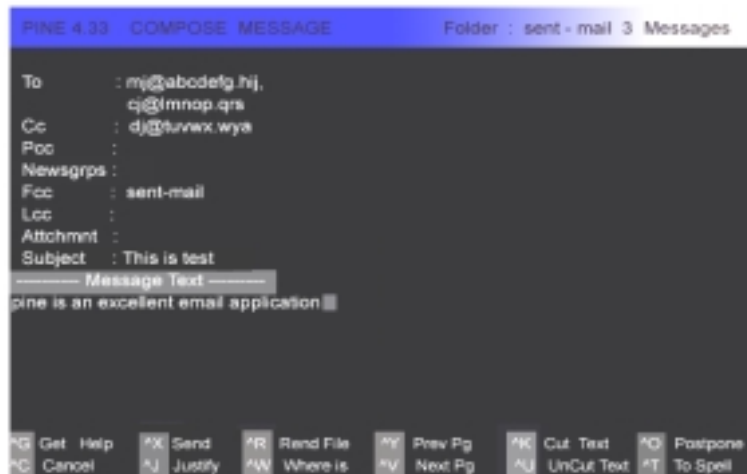



그림 24-18. 완성된 통보문

pico편집기

pine에서 통보문을 편집할 때 pico라고 부르는 통합된 본문편집기를 리용한다. 아래에 pico지령을 보여 주었다.

- Ctrl+V: 페지아래
- Ctrl+Y: 페지우
- Ctrl+K: 본문행자르기
- Ctrl+U: 본문자르기를 취소하기
- Ctrl+A: 행의 시작위치로 이행
- Ctrl+E: 행의 끝으로 이행
- Ctrl+J: 현재 단락(조절 및 배치)을 조절한다.

본문을 자르고 붙이자면 약간한 설명이 요구된다. Ctrl+K를 눌렀을 때 본문에서 한행이 없어 지면서 기억기의 완충기에 넣어 진다. 또한 련이어 잇달ㅏ 여러개의 행을 없애려고 Ctrl+K를 눌러도 잘리운 모든 행들은 완충기에 복사된다. Ctrl+U를 누르면 잘리웠던 문서가 다시 나타난다. 이때 복사되였던 완충기의 내용이 현재 유표의 위치에 복사되지만 완충기의 내용은 지워 지지 않는다. 이렇게 통보문본체안에 없애버렸던 본문을 다시 삽입할수 있다.

통보문의 응답과 넘기기 통보문에 응답하고 다른 수신자에게 그것을 넘기는것은 새로운 통보문을 조립하는 경우와 거의 같다. 해당한 통보문을 선택하거나 **R**를 리용하여 통보문에 응답한다(폴더색인에서 혹은 통보문보기에서). 이때 다음의 질문이 제기된다.

응답에 초기통보문도 포함되는가? 이 질문에 명백히 대답하면 전체 내용이 응답에 포함된다.

모든 수신자가 응답하는가? 이 질문은 하나의 통보문을 여러명의 수신자가 받을 때 제기된다(To마당 혹은 Cc마당에서). 응답으로 **Y**를 주면 통보문를 받을수 있는 모든 수신자들이 이 응답을 받아 들인다. 응답으로 **N**를 주면 통보문을 발송하는 오직 한사람만이 이 응답을 수신한다.

그림 24-19. 통보문에 대한 응답

```
PINE 4.33   COMPOSE  MESSAGE                Folder : sent - mail 3 Messages
To         : bb@cc.dd.ee.fff
Cc         :
Attchmnt   :
Subject    : This is a test (fwd)
---------- Message Text ----------
This is a message Forwarding another message
---------------- Forwarded message ----------------
Date:The,17,may 2001 10:05:00 -0400 (EDT)
From:Michael Jang <mj@localhost.localdomain>
To:mj@abcdefg.hij, cj@lmnop.qrs
Cc:dj@tuvwx.wya
Subject:This is test

pine is an excellent email application

^G Get Help  ^X Send     ^R Read File  ^Y Prev Pg  ^K Cut Text    ^O Postpone
^C Cancel    ^J Justify  ^W Where is   ^V Next Pg  ^U UnCut Text  ^T To Spell
```

그림 24-20. 통보문의 넘기기

다음으로 통보문구성화면은 알맞는 수신자, 제목행, 이미 준비된 어떤 본문으로 구성된다. 이 실례를 그림 24-19에서 보여 주었다.

새로운 통보문을 조립할 때 해볼수 있는데 이때 Ctrl+X를 누르면 응답을 전송하고 Ctrl+C를 누르면 응답을 취소한다.

통보문은 하가지 방법으로 전송한다. 즉 통보문을 선택하고 **F**을 리용한다. 이때 통보문합성창문이 통보문의 본문과 전송통보문이라는것을 지정하는 제목행과 함께 그림 24-20과 같이 나타난다.

▶미리 알아두기

이제는 두개의 가장 일반적인 인터네트조작 즉 Web을 열람하기 및 전자우편송수신방법을 파악하였으리라고 본다. 그렇지만 모뎀 즉 팍스조작에 대해서는 잘 모를것이다.

제 25장 《Linux 에 의한 팍스통신》에서는 Linux체계에서 송신 및 수신, 팍스의 인쇄를 지원하는 쏘프트웨어에 대하여 설명한다. 여기서는 응용프로그람에의 인쇄와 같은 방법으로 팍스전송방법과 자동적인 팍스전송체계설정방법에 대하여 고찰한다.

제25장. Linux 에 의한 팍스통신

우리는 이미 망접속을 실현하는 모뎀사용법에 대하여 배웠다. 모뎀사용법에서는 주로 인터네트에 접근하기와 거기에 리용되는 많은 봉사기능들에 대하여 보았다. 이 장에서는 팍스를 전송하거나 받을 때 리용하는 모뎀의 다른 리용방법을 고찰한다.

Linux 에는 여러가지 팍스쏘프트웨어프로그람묶음이 있다. Linux와 Unix 세계에서 기대되는 팍스프로그람묶음에는 자료를 전송할수 있는 지령행도구들과 팍스봉사를 자동적으로 진행할수 있는 프로그람을 비롯하여 여러가지 처리를 진행하는 많은 프로그람들이 포함되여 있다.

더우기 Linux의 많은 팍스프로그람묶음으로 일부 응용프로그람들의 인쇄특성을 리용하거나 사용자들이 전자우편을 보내는것과 같은 방법으로 팍스를 전송하는 지령행도구를 확장할수 있다.

이 장에서는 Linux의 팍스전송기술에 대한 폭 넓은 지식을 배울수 있으며 일부 전용팍스프로그람묶음을 리용하는 방법을 구체적으로 배우게 된다. 이와 함께 Red Hat Linux 7.1의 다른 배포물들에서 볼수 있는 전자우편팍스(efax), 공동팍스프로그람묶음의 리용방법에 대하여서도 자세히 배우게 될것이다.

주의	전자우편팍스는 이 책에 부속된 CD-ROM에는 없다. 만약 Red Hat Linux 7.1의 표준배포물이 있으면 이 프로그람묶음은 두번째 설치 CD 에서 찾아볼수 있으며 한편 http://www.redhat.com 주소 혹은 http://www.rpmfind.net주소와 같은 Web싸이트로부터 RPM파일들을 내리적재하는 방법으로 설치할수 있다.

Linux 팍스쏘프트웨어

일반적으로 Linux 세계와 Unix 세계에는 efax, NetFax, mgetty+sendfax, HylaFax와 같은 팍스들을 송수신할수 있는 몇가지 주요한 쏘프트웨어들이 있다.

이 모든것중에서도 efax는 일반적으로 구성 및 설치를 아주 쉽게 할수 있다. 왜냐하면 이 프로그람은 단일사용자체계에서 리용할수 있기때문이다. Linux를 가정에서 리용하는 많은 사용자들에게는 efax 가 적합하다.

물론 기초프로그람묶음으로서 efax는 다중사용자망환경에서 리용할수 있는 HylaFax와 NetFax와 같은 많은 고유한 특성들을 가지고 있지 못하다. 그러나 efax는 충분한 팍스전송능력을 가지고 있다. 이 팍스프로그람은 단일사용자 그리고 소규모사무처리사용자용으로 개발되였다.

Linux 를 위한 Efax

efax는 많은 Linux배포물(Red Hat Linux 7.1의 표준배포물을 비롯하여)을 가지고 있기때문에 디 팍토(de facto)표준쏘프트웨어라고 할수 있다. 이 장에서는 Linux의 전자우편팍스리용법에 중점을 두고 고찰한다.

일러두기	Linux의 또 다른 팍스프로그람묶음에 관심이 있다면 http://www.linuxapps.com/ 의 Linux Program Web싸이트의 목록을 보면 된다. 이 장에서는 팍스모뎀의 기능을 파악하는데 기본중점을 두고 고찰한다. Linux에서 모뎀을 어떻게 구성하는가를 설명하며 이미 모뎀이 설치되여 있는가를 확인하는 지도서를 보는 방법을 설명한다.

주의	이 장에서 론의되는 efax프로그람은 efax.com프로그람과는 관계 없다. 그것은 사실상 EdCasas에 기초한 《Ed's fax program》의 략자이다. http://www.cce.com/efax 주소를 찾아 보는 방법으로 efax에 대한 보다 많은 정보를 얻을수 있다.

Linux에 Efax의 설치

팍스의 기능을 높이고 체계에서 그것을 실행시키기 위한 첫 단계는 우선 체계에 efax쏘프트웨어가 있는가를 확인하는것이다. 만일 이 책과 함께 제공된 Linux의 판본을 체계에 설치했다면 이미 그 쏘프트웨어는 설치되였을것이다. rpm지령으로 이것을 검사할수 있다. 즉

$ rpm -q efax

efax 가 설치되였다면 efax-0.9-8과 같은 efax프로그람묶음이 화면에 표시된다. efax 가 설치되지 않았다면 그때 rpm 지령은 다음과 같은 간단한 통보문을 현시한다.

Package efax is not installed.

efax가 설치되지 않았다면 그때 두가지 선택으로 진행할수 있다.

- 배포물의 설치CD로 그것을 설치한다. Red Hat Linux 7.1의 표준판본이 있다면 두번째 설치 CD-ROM 에 존재하게 된다.
- 인터네트로부터 그것을 내리적재하고 설치한다.

Red Hat CD-ROM으로 Efax 의 설치

표준CD-ROM프로그람묶음으로 Red Hat Linux를 설치하고(이 책에 부속된 CD에는 없다.) 배포물에 따라 들어 온 efax의 판본을 설치하려 한다면 mount지령을 리용하여 표준경로 /mnt/cdrom에 Red Hat Linux 7.1설치 CD-ROM을 설치해야 한다(이 실례에서는 CD-ROM이 장치 /dev/cdrom에 있다고 가정한다.). 즉

```
# mount /dev/cdrom /mnt/cdrom
```

일단 CD-ROM이 설치되면 rpm지령을 리용해서 efax-0.9-8.i386.rpm파일을 설치한다. 즉

```
$ rpm -i /mnt/cdrom/RedHat/RPMS/efax-0.9-8.i386.rpm
```

인터네트에서 Efax를 내리적재하고 설치

만일 efax의 최신판이 요구된다면 Web싸이트주소 ftp://metalab.unc.edu /pub/Linux/apps/serialcomm/fax/로부터 얻을수 있다. 이 쏘프트웨어의 최신판본은 0.9이며 내리적재된 파일이름은 efax-09.tar.gz이다.

일단 이 파일을 내리적재했다면 /tmp와 같은 림시등록부에 이 압축파일을 풀어 넣기 위하여 tar지령을 리용한다. 즉

```
$ tar xzvf efax-09.tar.gz
```

다음은 현재 작업등록부를 림시등록부로 바꾸고 (이 경우에는 지령 cd /tmp/efax-0.9을 리용한다.) 뿌리사용자로 가입하여 다음과 같은 지령을 실행시킨다.

```
$ make
```

이 지령을 리용하여 efax 쏘프트웨어를 콤파일할수 있다. 콤파일과정에 오유가 없다면(efax는 안전한 프로그람이다.) 다음의 지령을 리용하여 기정등록부에 efax프로그람을 설치할수 있다.

```
$ make install
```

efax쏘프트웨어프로그람에 대한 콤파일과 설치방법을 구체적으로 알려면 배포물에 들어 있는 README파일을 참고해야 한다.

주의 Red Hat Linux7.1에서 **make install**지령을 리용하기전에 **Makefile**의 내용을 일부 변경해야 한다. 이를 위하여 favorite 본문편집기에서 **Makefile**파일을 연다. 이 파일은 efax-09.tar.gz파일(/tmp/efax-0.9)의 등록부에 있다. 이 파일안의 **install**부분에서 두번째 지령은 **cp fax.1 efax.1 efix.1 $(MANDIR)/man1**로 되여 있다. 이것을 그림 25-1에서 보여 준것처럼 **cp fax.1 efax.1 efix.1 /usr/share/man/man1**로 변경한다.

Linux의 Efax구성

첫 팍스를 보내기전에 먼저 efax의 일부 파라메터들을 규정해야 한다. 이를 위하여 /usr/bin/fax파일을 편집하고 확실한 정보들을 설정해야 한다.

- FROM으로 시작되는 행을 찾아서 전화번호를 설정한다.(FROM="+1 212 55 1212")
- PAGE로 시작되는 행을 찾아서 매 나라의 고유한 페지규격을 설정한다(PAGE=letter,PAGE=legal,혹은 PAGE=a4.).

```
all: efax efix

efax: efax.o efaxlib.o efaxio.o efaxos.o efaxmsg.o
      $(CC) -o efax $(LDFLAGS) efax.o efaxlib.o efaxos.o efaxmsg.o
      strip efax

efix: efix.o efaxlib.o efaxmsg.o
      $(CC) -o efax $(LDFLAGS) efax.o efaxlib.o efaxmsg.o
      strip efax

install:
      cp fax efax efix $(BINDIR)
      cp fax.1 efax.1 efix.1 /usr/share/man/man1

clean:
      rm -f efax efix efax.o efix.o efaxlib.o efaxiio.o efaxos.o efaxmsg.o

efax.o:     efax.c     efaxmsg.h    efaxlib.h    efaxio.h    efaxos.h
efaxio.o:   efaxio.c   efaxmsg.h                 efaxio.h    efaxos.h
efaxos.o:   efaxos.c   efaxmsg.h    efaxlib.h                efaxos.h
efix.o:     efix.c     efaxmsg.h    efaxlib.h
efaxlib.o:  efaxlib.c  efaxmsg.h    efaxlib.h
```

그림 25-1. 편집된 efax의 Makefile파일

- DEV로 시작되는 행을 찾아서 그 장치가 modem 혹은 ttyS0과 일치하는가를 확인한다. 모뎀장치에 대한 보다 구체적인 자료는 제18장에서 참고한다.
- NAME으로 시작되는 행을 찾아서 팍스를 받는 수신자이름을 써넣는다.

파일을 편집할 때 하쉬표식(#)으로 시작하는 행이 설명문이라는것을 명심하고 구성에 영향이 미치지 않도록 한다. 하쉬표식이 있는 행에 대하여 설명하고 기정구성에로 쉽게 절환할수 있게 행에 변경을 추가할수 있다.

일러두기	북아메리카의 밖에서 팍스를 전송하려면 **TELCVT**변수가 들어 있는 **/usr/bin/fax** 구성정보들을 찾아서 전화망의 민족코드를 표현하는 **TELCVT**에 설명문이 붙었는가를 확인해야 한다.

등록부준비

마지막단계는 팍스를 받거나 보내기전에 spool, log, lock등록부 (/var/spool/fax, /var/log/fax와 /var/lock)들을 따로따로 준비하는것이다.

우선 팍스보조등록부들이 거기에 있다면 ls /var/spool을 리용하여 등록부들이 존재하는가를 검사한다. 팍스설치과정에 이 등록부가 만들어 져서 존재하게 된다. 우와 같은 방법으로 /var/log등록부에 대한 처리를 반복한다.

다음은 제22장 《Linux를 인터네트에 접속》에서 KPPP로 수행한것처럼 임의의 사용

자가 팍스 완충등록부를 호출할수 있도록 등록부허가를 설정한다.

$ chmod u+s /var/spool/fax

fax log등록부에 대한 처리도 같은 방법으로 반복한다.

$ chmod u+s /var/log/fax

마지막으로 lock파일등록부에 대한 호출권한은 모든 사용자들에게 다 추가된다. 그러나 팍스를 리용할 때 누구나 동시에 팍스를 리용할수는 없다.

$ chmod 777 /var/lock

첫 팍스발송

팍스프로그람묶음은 세개의 주요프로그람 efax, efix, fax로 이루어 졌다.

- efax프로그람은 팍스체계프로그람의 핵심이다. 이 프로그람은 팍스를 표준 팍스파일형식 즉 TIFF파일을 압축한 Group 3으로 보내거나 받을수 있다. Group 3형식으로 변환되지 않은 파일이나 본문파일, 인쇄파일(PostScript)들은 efax프로그람으로 전송할수 없다.
- efix프로그람은 모든 파일들을 본문, bitmap영상, TIFF팍스형식으로 변환할 때 리용한다. efax와 efix를 결합하는 방법으로 많은 image들과 본문파일들을 팍스로 전송할수 있다.
- fax프로그람은 팍스의 만들기, 송신, 수신, 보기, 팍스인쇄와 같은 종합적인 팍스통합환경을 지원하는 프로그람이다. 이 프로그람은 efax와 efix, 완전한 Linux팍스기능들을 지원하는 Linux의 팍스프로그람들로 통합되여 있다.

efax프로그람에 의한 팍스송수신방법을 알기 위해 먼저 팍스의 모든 기능들을 다 제공해 주는 팍스프로그람에 중점을 두고 고찰한다.

팍스발송

일반적으로 팍스프로그람들은 본문파일 혹은 인쇄파일(PostScript)들을 팍스로 쉽게 전송한다. 이에 대한 팍스지령은 다음과 같다.

$ fax send *options number file*

3개의 선택항목을 리용하여 팍스를 전송할수 있다.

-l 저해상도리용(96점/인치)

-v 긴 통보문과 프로그람상태제공

-m 전화번호는 이미 다이얄되여 있다고 가정한다(이 경우에는 전화번호를 돌리지 않는다.).

전화번호는 전화번호를 돌릴 때 리용하는 전화번호형태를 그대로 쓴다. 실례로 국부전화번호 555-1212를 돌린다면 전화번호는 5551212 혹은 555-1212의 형식으로 돌려야 한다. 만일 지령행전화번호가 9로 시작될 때는 95551212,9-5551212나 9-555-1212의 형식으로 돌려야 한다.

보다 더 구체적인 실례를 보자. 등록부 /tmp안에 textfile이라고 하는 본문파일이 있고 지령행전화를 호출하는 수자 9를 리용하여 전화번호 123-4567을 돌려 저해상도팍스로서 그 파일을 보내려 한다고 하자. 이에 대한 지령은 다음과 같다.

```
$ fax send -l 9-123-4567 /tmp/textfile
```

이때 보내려는 본문파일의 내용은 다음과 같다.

```
FAX TRANSMISSION:
TO: Arman Danesh
FROM: Arman Danesh
NOTE:

This is a test of the efax package. Using the fax command, we can send text files
as faxes.
```

이것을 실행하면 그림 25-2와 같은 팍스가 전송된다.

```
FAX TRANSMISSION:

TO: Arman Danesh

FROM: Arman Danesh

NOTE:

This is a test of the efax package. Using the fax command, we
can send text files as faxes.
```

그림 25-2. efax팍스실례

다중파일을 하나의 팍스로 전송

팍스로 한개 파일을 보내는것외에 하나의 팍스로서 여러개의 파일들을 팍스전송할수 있다. 실례로 /tmp/textfile1파일의 내용과 /tmp/textfile2파일의 내용을 동시에 전송하려 할 때 다음의 지령을 리용한다.

```
$ fax send 9-123-4567 /tmp/textfile /tmp/textfile2
```

이때 파일의 전송순서는 textfile, textfile2이다.

주의 이 방법으로는 여러개의 본문파일들만 결합할수 있다. 이 처리로는 본문파일들과 인쇄파일 혹은 인쇄파일들과 인쇄파일들을 결합하지 못한다.

그러나 파일의 페지들을 순서대로 차례차례 보내거나 그 일부분만을 보내려고 한다면 우선 파일들을 팍스양식의 TIFF파일로 변환해야 한다. 팍스의 매 페지는 그때 분리된 파일에 배치되고 파일에서 페지들을 따로따로 선택해서 팍스전송할수 있다.

여기서 팍스프로그람의 팍스양식의 파일작성기능을 리용할수 있다. 실례로 세페지로 된 /tmp/psfile인쇄파일을 가지고 있다면 다음의 지령을 리용하여 그것들을 세개의 팍스양식 TIFF파일로 변환할수 있다.

```
$ fax make /tmp/psfile
```

결과페지들은 파일이름이 각각 /tmp/psfile.001, /tmp/psfile.002, /tmp/psfile.003

이라고 하는 세개의 분리된 파일속에 포함된다.

파일을 팍스양식으로 변환할 때 결과페지들은 파일확장자가 순서대로 된 세개의 수자를 가지는 같은 이름들로 만들어 진다. 실레로 test.txt파일은 test.txt.001, test.txt.002 등의 이름으로 불리우는 팍스페지들을 만든다.

fax make지령을 리용할 때 리용가능한 선택항목은 저해상도방식의 –l 선택항목을 리용하는것이다.

```
$ fax make –l /tmp/psfile
```

먼저 변환된 세개의 페지문서로 돌아 가 보자. 세번째 페지를 먼저 전송하고 다음에 첫 페지를 전송하며 두번째 페지는 전송하지 않으려면 다음과 같은 지령을 리용해야 한다.

```
$ fax send /tmp/psfile.003 /tmp/psfile.001
```

인쇄방법에 의한 팍스전송

팍스지령을 리용하여 팍스를 보내는것은 많이 류행되고 있지만 Netscape나 WorldPerfect 혹은 ApplixWare Words와 같은 응용프로그람으로 전송할 때는 그리 리용되지 않는다.

주의 물론 수중에 하나밖에 없는 모뎀을 Netscape 에서 인터네트를 열람하는데 리용하고 있다면 인터네트접속에 모뎀을 리용하는 동안은 Netscape에서 팍스를 전송할수 없다.

실레로 Netscape를 보자. Netscape에서 파일을 인쇄할수 있으며 인쇄된 파일은 PostScript파일을 만든다. 팍스에 의한 페지전송은 팍스에 의한 파일전송지령행에로 이행할것을 요구하면서 그 프로쎄스에 두번째 단계를 추가한다.

그러나 이것은 Windows 나 Macintosh응용프로그람으로 팍스를 전송하는 방법과는 류사하지 않다. 여기서 일반적으로 팍스전송은 인쇄방법과 류사하다.

다행히도 팍스로 인쇄한 파일을 전송하기 위한 efax를 리용하는 Linux에서 특수한 인쇄장치를 만들수 있다.

처리방법은 매우 단순하다. lpd인쇄체계에서는 특수화 된 입구려파프로그람을 리용할수 있다. 만일 입구려파프로그람을 인쇄대기렬에 받아 들였다면 대기렬로 전송하려는 파일은 대기렬에 배치되기전에 입구려파프로그람에 먼저 전송된다. 만일 려파프로그람이 그 어떤 자료도 복귀시키지 못하고 파일들을 팍스로 전송한다면 인쇄기에서는 그 어떤 작업도 수행할수 없게 된다.

팍스모뎀으로 인쇄하기 위한 인쇄기대기렬은 세가지 단계로 설정할수 있다.

1. 인쇄대기렬구성
2. faxlpr스크립트설정
3. 입출력완충등록부준비

이 모든 준비는 체계에서 뿌리사용자로서 집행할것을 요구한다.

인쇄대기렬구성

인쇄대기렬을 설정하려면 /etc/printcap.local 파일을 편집해야 한다. 행인쇄기데몬 lpd(Line Printer Daemon)을 재기동할 때 이 정보는 표준 /etc/printcap 인쇄구성파일 안에 병합된다. 선택된 본문편집기로 /etc/princap,local파일을 열고 다음의 항목들을 입력한다.

```
fax: \
:sd=/var/spood/fax: \
:mx#0: \
:lp=/dev/null: \
:if=/usr/bin/faxlpr:
```

이 항목들은 팍스대기렬의 이름을 기입하고 /var/spool/fax에 입출력등록부를 설정한다. 그리고 mx#0로 작업크기를 제한이 없게 설정하며 /dev/null 에 인쇄장치를 설정하고 /usr/bin/faxlpr등록부에 입구려파프로그람을 설정한다.

주의 제18장에서 론의된 Apsfilter도구만을 리용하고 제15장에서 론의된 Printconf를 리용하지 않는다면 직접 /etc/printcap파일을 편집할수 있다.

faxlpr스크립트설정

팍스프로그람은 실지 매우 유연한 쉘스크립트이다. 스크립트를 자체의 규정된 이름(fax)으로 호출하면 지령이 지령행에 입력되고 그 지령에 따라서 처리된다고 가정한다. 스크립트를 대용이름(faxlpr)으로 부르면 그 스크립트를 lpd인쇄기대기렬을 통하여 팍스를 전송하는데서 제기된 질문이라고 가정한다.

이를 위하여 팍스구성파일을 련결할 필요가 있다. 실례로 그것이 /home/mj등록부에 있는 .efaxrc이라면 다음과 같은 지령을 리용한다.

```
$ ln / home/mj/.efaxrc /usr/bin/faxlpr
```

home등록부의 .efaxrc파일을 설정하지 못했다면 직접 팍스지령을 련결할수 있다.

```
$ ln /usr/bin/fax /usr/bin/faxlpr
```

마지막으로 lpd데몬을 재시동한다. Red Hat Linux 7.1에서 /etc/rc.d/init.d/lpd재시동지령을 리용한다. 다른 배포물들은 서로 다른 등록부에 lpd데몬을 보관하고 있다.

인쇄대기렬을 통한 팍스전송

일단 이 모든것을 수행하면 lpd체계를 리용하여 첫 팍스를 보낼 준비가 완성된다. lpr지령을 리용하여 파일을 인쇄하는것과 같은 방법으로 본문이나 인쇄(PostScript)파일들을 팍스전송할수 있다. 팍스전송을 위하여 리용되는 새로운 표기법은 전화번호를 규정하는 lpr지령의 -J를 리용하는것이다.

실례로 fax라고 부르는 인쇄기대기렬을 리용하여 전화번호 555-1212에로 팍스전송하게 하는 /tmp/testfile 파일이 있다면 다음의 지령을 리용할수 있다.

```
$ lpr -Pfax -J 5551212 /tmp/testfile
```

lpd데몬은 파일을 입구려파프로그람(faxlpr)으로 통과시키는데 이것은 이 파일을 팍스로 전송한다. 팍스 lpr는 처리의 성공 혹은 실패를 표시하는 상태정보를 팍스전자우편으로 사용자에게 보낸다.

lpr지령에서의 특수한 선택항목을 규정하고 PostScript출력을 산생시키는 그 어떤 프로그람에서도 이 방법으로 팍스내용을 인쇄할수 있다(PostScript만이 아닌 Linux의 Netscape 6와 WordPerfect, StarOffice도 제한없이 리용할수 있다.).

팍스수신

팍스수신은 아주 간단한 처리이다. 팍스는 3가지 방법 즉 수동적인 방법과 규정된 가입호출에서의 자동설정방법, 모든 가입호출에서의 자동설정방법으로 수신할수 있다.

수동적인 팍스수신

수동적으로 팍스를 수신하기 위해서는 전화가 울리면 전화로 대답하고 팍스를 수신하도록 모뎀에게 전화번호를 알려 주어야 한다. 이것은 팍스지령의 receive선택항목으로 수행한다.

전화선이 련결되고 팍스로 들어 오는 전화번호를 안다면 간단하게 다음의 지령을 줄수 있다.

```
$ fax receive
```

fax receive지령에 의해서 팍스로 들어 오는 파일이름을 규정할수 있다. 팍스의 매 페지는 순차적으로 수자화된 파일확장자(001,002,003등)를 가진 분리된 파일에 보관된다. 실례로 세페지의 팍스를 수신하려면 다음과 같은 지령을 리용할수 있다.

```
$ fax receive testfax
```

수신된 팍스는 testfax.001, testfax.002와 testfax.003라고 하는 파일에 보관된다. 수신된 팍스를 보려면 이 절의 마지막에서 서술하는 fax view 지령을 리용해야 한다.

들어 오는 팍스의 파일이름을 규정하지 않았다면 그때 fax지령은 팍스를 받은 날자와 시간으로 된 10개 수자로 된 파일이름을 만든다. 실례로 12월 24일 오후 9시 52분 35초에 수신된 팍스는 1224095235.001,1224095235.002, 1224095235.003 등으로 된 파일로 보관된다.

팍스파일은 기정 /var/spool/fax등록부에 보관된다. fax queue지령으로 팍스 spool 등록부의 내용을 볼수 있다.

```
$ fax queue
```

Fax files in /var/spool/fax :

```
-rw-r- r--1   root   247         Jul   2  17:49 0702174857 .001
-rw-r- r--1   root   247         Jul   2  17:59 0702175842 .001
-rw-r- r--1   root   247         Jul   4  17:14 0704171330 .001
```

이 팍스들을 보려면 X Windows에서 fax view지령을 주어야 한다. 즉

```
$ fax view 0704171330.001
```

이 지령은 팍스를 그림 25-3 에서 보여 준것처럼 려파하고 xloadimage프로그람으로 현시한다.

일러두기	사용자들은 /usr/bin/fax 스크립트에서 VIEWCMD변수를 조종하는 방법으로 팍스려파프로그람을 변경할수 있다.

인쇄는 fax print지령을 리용하여 간단히 할수 있다.

```
$ fax print 0704171330.001
```

이 지령은 보통 lp라고 하는 기정인쇄기대기렬상에서 같은 fax내용을 인쇄한다.

규정된 호출에 대한 자동적인 팍스수신

수신하는 다음전화번호가 들어 오는 팍스라고 한다면 팍스프로그람이 호출대기, 전화로 대답하기, 팍스를 수신하기, 정지 및 끝내기하도록 설정할수 있다. 이를 위해 fax 지령의 answer선택항목을 리용한다. 즉

```
$ fax answer
```

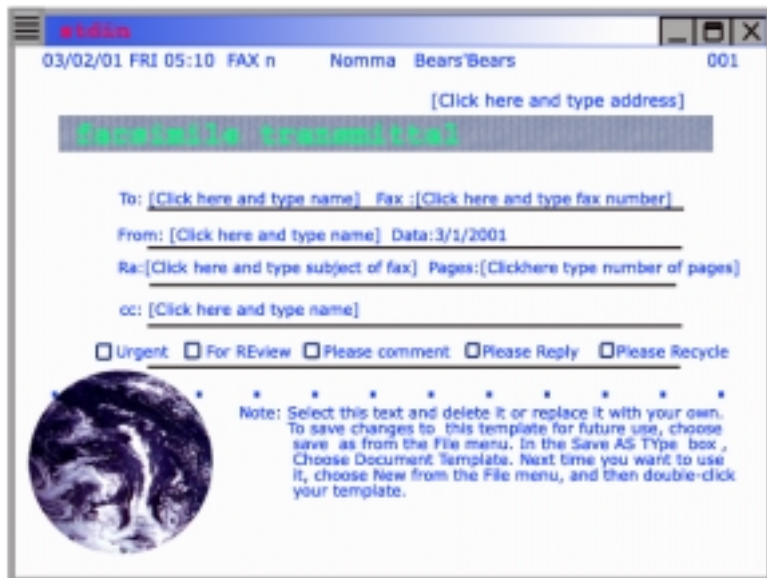



그림 25-3. xloadimage에 현시된 팍스

fax프로그람을 기동되면 전화가 울릴 때까지 기다렸다가 두번째로 울린후에야 대답한다. fax receive지령으로 들어 오는 팍스를 파일이름을 산생한 날자로 보관한다. 들어온 팍스를 fax view와 fax print지령으로 보거나 인쇄할수 있다.

들어 오는 모든 팍스에 대한 자동적인 팍스수신

팍스모뎀이 들어 오는 팍스들에 리용하는 전화선과 접속하였다면 송신호출을 규정한 조건에서 들어 오는 팍스에 대하여 련속적으로 관리하는 efax 프로그람을 실행시킨다.

팍스지령의 -w 선택항목을 리용하여 팍스모뎀으로 들어 오는 호출을 대기하는 데몬으로 프로그람을 실행시킨다. 전화지령행에서 대표적인 efax 지령은 다음과 같다.

```
$ efax -d /dev/modem  -w  -iSO=5
```

선택항목 -d는 efax가 모뎀장치파일 /dev/modem라는것을 가리킨다. 선택항목 -w는 팍스가 작업상태(표준U.S전화번호《돌리기》)를 대기하도록 한다. 선택항목 -iSO는 작업상태로 되기전에는 다섯번 전화번호를 돌릴 때까지 기다리게 한다.

체계를 팍스가 임의의 시간에도 응답하게 하려면 Linux를 설치할 때마다 데몬이 기동하도록 초기기동파일 /etc/rc.d/rc.local 의 efax -d/dev/modem -w -iSO=5지령을 주어야 한다.

주의	rc.local 초기기동파일의 경로는 Linux 배포물마다 다를수 있다.

데몬의 응답을 정지시키려면 다음의 지령을 리용하여 데몬을 금지시켜야 한다.

```
$ efax  stop
```

일단 efax의 비접속통보문이 나타나면 모든 자동적인 응답을 끝낸다.

▶미리 알아두기

지금까지는 Linux에 의해서 팍스를 보내고 받는 과정을 보았다.

이 책의 다음부분은 완전히 새로운 부분 즉 SOHO환경에서의 Linux 이다. 여기에서는 망환경에서 Linux를 적재하는 방법과 기본망구성을 시험하는 방법을 배우게 된다. 그때 Linux의 SOHO환경에서 리용할수 있는 봉사형태들을 구체적으로 본다. 뿐만아니라 전자우편, 인트라네트 Web봉사기, Unix를 위한 파일봉사기, Windows와 Novell망을 고찰한다.

제 6 편. 소규모사무/가정사무에서 Linux 의 리용

제 26 장. SOHO 에서 Linux 를 어디에 리용하는가

이제는 이 책의 초점을 변경시키려고 한다. 이제부터는 개별적인 워크스테이션상에서의 Linux 기동에 대한 확고한 리해를 가지고 가정에서 인터네트에 접근하여 Web 를 경유함으로써 전자우편 및 전자편지들을 작성하는것과 같은 작업을 수행하도록 하는 우수한 조작체계인 Linux 를 쓰는 방법을 보기로 한다.

Linux 의 실지 본바탕은 의연히 망세계에서 나타난다. 망에 결합시킬 두대의 PC 를 가지고 집에서 소규모적인 사무작업을 진행하는 경우를 보자. Linux 는 많은 망들에서 하나의 큰 파일 및 인쇄봉사체계로 된다. 사무작업환경으로서 인트라네트망을 구성하려고 한다고 하자. Linux 는 위력한 Web 및 자료기지봉사체계를 제공함으로써 비용을 적게 들이면서 인트라네트환경에로 사무작업을 유도하는 한가지 수단을 제공한다.

그 PC 들로 이루어 진 인트라네트망을 인터네트망에 접속하려고 할 때 우수한 경로기의 하나로서 Linux 를 선택할수 있고 많은 경우 그 인트라네트망을 세계망과 접속시킬 때 경로기로 Linux 를 설치한 콤퓨터를 리용하면 하드웨어적인 경로기를 구입하여 리용하는것보다 훨씬 원가가 적게 든다.

끝으로 가정이나 사무작업망들에서 흔히 제기되는 능력이나 안정성에 대한 문제들을 Windows 에 기초한 탁상환경에서 해결할수 없을 때 워크스테이션조작체계로서의 Linux 가 이러한 문제들에 훌륭히 대처할수 있는것으로 하여 그 우월성이 발현된다. 시끄러운 절차가 없이도 Linux 는 서로 다른 망조작체계들인 Unix, Windows, Novell 들과 결합되여 Windows 와 Unix 응용프로그람들을 모두 호출할수 있게 하며 그밖의 기타 파일들과 자료를 쉽게 공유할수 있게 한다.

이 장에서는 소규모사무/가정사무(일반적으로 SOHO 라고 부른다.)작업에서 Linux 조작체계를 리용하는 방법을 취급하는데 이 조작체계를 리용하면 사무작업을 진행하는데 드는 원가는 훨씬 눅어 진다.

파일봉사기로서의 Linux

소규모망에서 가장 초보적으로 요구되는것이 파일봉사체계이다. 파일봉사체계들은 모든 사용자들이 필요에 따라 공유파일들을 호출하게도 하고 파일들을 예비로 보관하도록 그에 해당한 공동기억장소를 제공해 준다(따라서 망에 련결된 모든 워크스테이션들은 제각기 매일 자체의 국부적인 장소에 파일들을 예비로 가지고 있을 필요 없게 된다).

Windows 나 Novell 망에서 보통 다음과 같은 두가지 방법중 한가지 방법으로 이 작업을 진행한다.

- 공동파일봉사체계의 리용
- 동위망의 리용

공동파일봉사체계

공동파일봉사체계는 망에서 어느 한 워크스테이션에만 해당되는 체계로가 아니라 통일적인(중앙)파일봉사체계로서 쓰인다. 망에 련결된 워크스테이션이나 콤퓨터들을 리용하는 사용자들은 마치 자기콤퓨터의 등록부나 디스크구동기처럼 그 파일봉사체계의 등록부들을 호출할수 있다.

공동파일봉사기들은 공유자료와 여분화처리를 관리하는 우점을 가지고 있다. 그러나 이 방식은 동위망을 리용하여 작업하는것보다 원가가 많이 든다. 또한 이 방식에서는 권한이 없는 사용자들이 망탕 정보들을 호출할수 없도록 기밀적으로 핵심적이며 공유적인 자료에 대한 호출조종수단도 설정할수 있는 특권을 베풀고 있다.

Linux 는 원가가 적게 드는 종합적인 파일봉사체계로 쓰일수 있다. 또한 널리 쓰이는 Windows 95/98/Me 보다 Linux 의 안정성과 다중과제처리능력이 오히려 더 좋기때문에 든든한 파일봉사체계가 요구될 때는 보통 Linux 체계를 리용한다.

제 29 장 《Red Hat Linux 7.1 을 Window 와 Novell 망에 통합》에서 Linux 를 파일봉사체계로 설정하는 방법을 보게 될것이다.

동위망에서의 파일공유

동등한 권한을 가진 망에서는 중심파일봉사체계가 따로 없다. 대신에 매 사용자들은 자기들의 PC 에 있는 등록부와 구동기중에서 어느것을 망에서 공유하여야 하며 어느 준위의 호출조종이 불가능한가를 결정해 준다. 이 방식에서는 망에 련결된 모든 워크스테이션과 콤퓨터가 각각 작은 파일봉사체계로 된다.

이 방식은 사용자들이 공유할 자료가 어느것이며 또 누구와 공유해야 하는가를 결정할수 있는 유연성을 띠고 있지만 결함들도 있다. 즉 파일공유 및 호출조종수단들에 대한 통일적인 통제가 없고 중요한 자료를 여분으로 보관해 주는 통일적인 장소가 없으며 망에 련결된 다른 개별적인 사용자들이 어떤 PC에 있는 공유파일들을 망탕 호출하게 됨으로써 경우에 따라 그 PC 의 탁상에서 실행충돌이 있게 되는 결함들이 있다.

Linux 는 여러 형태의 망들과 공유할수 있는 능력을 가진것으로 하여 동위망파일봉사체계로 동작할수 있다. 그러나 Linux 의 다중과제처리능력은 여러 사용자들이 다른 작업중에 있는 PC 의 파일들을 호출할 때 제기될수 있고 많은 실행충돌을 완화시켜 준다.

결국 작은 망에서 원가가 적게 드는 Linux 에 기초한 공동파일봉사체계를 적용하면 더욱 매력적인 공동파일봉사체계로 될수 있으며 따라서 관리 및 통일적인 여분화처리에서 많은 리익을 볼수 있다.

인쇄봉사기로서의 Linux

파일봉사체계뿐만아니라 인쇄봉사체계로도 망이 형성된다. 보통 공동파일봉사체계는 인쇄봉사체계의 역할도 수행한다.

같은 망에서 일반적으로 공동방식과 동등방식을 모두 지원하는 인쇄봉사체계를 찾을수 있다. 인쇄기를 리용하는 사용자들은 자기 책상에서 자기가 가지고 있는 인쇄기들을 리용하지만 인쇄기가 없는 사용자들은 공동령역에 국한된 인쇄기를 공유할수 있다. 공동인쇄기를 리용하는 사용자는 망을 경유하여 인쇄기를 공유하므로 동등하게 공유된 인쇄기보다 공동인쇄기에서 직접 문서들을 인쇄할수 있다.

물론 Linux 는 두 환경에서 다 작업할수 있다. Linux 는 PostScript 및 HP 와 호환할수 있고 PCL 인쇄기를 비롯한 안정성이 좋은 인쇄기들을 지원한다. 또한 Linux 의 인쇄기공유선택항목들은 Unix/Linux, Windows, Novell 망에 있는 인쇄기들을 공유할수 있는 능력을 가지고 있다.

제 29 장에서는 Linux 를 인쇄봉사체계로 설정하는 방법을 고찰한다.

자료기지봉사기로서의 Linux

Unix 가동환경과 마찬가지로 Linux 도 PostgreSQL 및 mSQL 을 비롯하여 넓은 분야의 자료기지프로그람제품들을 가지고 있다.

1998 년에 들어 서면서 전문상업자료기지판매업체들은 자료기지판매시장에서 Linux 용제품들을 전시하고 공개하는 방법으로 Linux 를 후원하였다. 실례를 들면

- Read Hat 는 Red Hat Linux 7.1 용으로 선택된 PostgreSQL 7.1.2 을 통합한 Red Hat Database 작성과정을 공개하였다. 이 자료기지는 시장을 독점하려는 큰 규모기업용자료기지체계들에 대항하여 중간급규모의 사무단위들과 개별적인 사무단위들을 목적으로 하여 나왔다.
- Oracle 는 Linux 2.4 핵심용으로 선택된 자료기지/응용프로그람체계도구들중에서 9i 묶음을 2001 년 초에 공개하였다. 이것은 가장 웃준위의 Unix 체계들로부터 큰 범위의 업무용자료기지들을 Linux 에 이식하는것을 담보한다.
- Sybase PowerBuilder 는 Linux 상에서 SQL Anywhere database Studio 응용프로그람을 지원한다. 이 프로그람제품의 원가는 WindowsNT/2000 프로그람제품의 절반으로서 문서화된 실례들을 가지고 있다.
- IBM 은 Linux 를 위하여 자기의 DB2 Universal Database 를 공개하였으며 Windows 2000/NT 체계를 비롯하여 제일 웃준위의 Unix 체계들의 자료기지들을 사용자들이 Linux 에 이식하는것을 장려한다.

Linux 가 인기를 끄는 파일봉사체계 및 Web 봉사체계로 제공되였기때문에 표준적으로 널리 쓰이는 관계형자료기지체계들을 리용하는 조직체들에서 든든하면서도 신축성이 있는 자료기지가동환경으로 전개할수 있다.

후에 이야기되지만 Linux 가 설치된 Web 봉사체계에 Linux 에 의존한 자료기지를 결합하면 Linux 가 완전한 인트라네트봉사체계로 봉사할수 있다.

인트라네트봉사기로서의 Linux

오늘날 정보화시대에서 주목되고 있는것은 인트라네트망이다. 인트라네트망은 어떤 조직체내에서 정보들과 응용프로그람들을 공유하기 위하여 TCP/IP, Web 열람기, 인터네트표준전자우편과 같은 인터네트기술을 활용하는 기업들의 내부에서 널리 쓰이는 망이다.

인트라네트와 그것을 전개하는데 리용되는 하드웨어 및 쏘프트웨어의 언급에서는 큰 기업들과 조직체들에만 적용되는 비용이 큰 인트라네트만을 소개한다. 그러나 이것은 더 허용되지 않을것이다.

소규모적인 사무분야에서도 인트라네트는 Web 열람기를 통하여 모든 사무원들이 읽어 볼수 있게 정보를 공개하는 편리한 수단을 제공할수 있다. 앞으로는 비용이 허용되는

범위에서 모든 사람들이 전자적이며 직결적인 Web 에 기초한 형태들을 도입하여 사무작업에서 종이문서작업을 없앨수 있다.

또한 조직체를 거쳐 접수한 소규모적인 자료기지 즉 각이한 사용자들이 사용한 일반기지들을 가지고 있을 때 인트라네트는 사용자들이 그 자료를 작성하여 보관하고 갱신하기 위하여 자료기지도구들을 호출하지 않고도 그 정보를 호출할수 있도록 일반적인 방법을 제공할수 있다. 인트라네트 Web 체계에 자료기지를 통합하면 Web 에 기초한 형태들은 자료기지를 참고하여 쓰일수 있고 완전한 기능적인 관계형자료기지체계를 비롯하여 강력하면서도 유연성이 좋은 Web 봉사체계들을 충분히 보장하는 Linux 는 Oracle 및 값 비싼 자료기지통합도구와 같은 상업용자료기지인 Windows NT/2000 을 전개하는데 드는 비용을 없애고 인트라네트봉사체계를 만들수 있게 한다. 또한 기업용자료기지들을 사용할수 있게 하는 인트라네트프로그람들과 응용프로그람들을 쉽게 생성하려는 Linux 무료형태의 도구들도 있다.

제 32 장 《Web 봉사기구성》에서는 Linux 에 기초한 인트라네트봉사체계를 설치하고 실행시키기 위한 기초들을 설명한다.

응용프로그람봉사기로서의 Linux

Linux 는 원래 파일/인쇄/인트라네트봉사체계와 응용프로그람봉사체계로 동작하도록 설계되여 있고 Telnet 와 같은 X Windows 규약이나 말단접속을 사용하여 응용프로그람봉사체계의 망들에서는 탁상에서 응용프로그람을 실행하며 자료는 그 봉사체계와 호출된 장소에 각각 보관된다.

일부 응용프로그람들에 대하여 그리고 일부 정황에서 응용프로그람들의 실행을 집중화시키면 성능 및 관리적인 측면에서 유리할수 있다. 만약 Windows 나 Novell 망을 운영하고 있다면 Linux 는 여러가지 목적에 따라 응용프로그람봉사체계로 동작할수 있다.

- 강력한 Unix/Linux 우편쏘프트웨어와 같은 문자에 기초한 프로그람들의 실행
- 집중화된 Linux 자료기지에 대하여 전용으로 설계되는 문자에 기초한 프로그람들의 실행
- Unix/Linux 쉘에서 실행하도록 설계한 기업용전화번호수첩과 같은 기업전용 응용프로그람의 호출

만일 Windows 탁상우에 X 봉사체계를 설치하면 Linux 응용프로그람봉사체계에서 응용프로그람들이 실행하는동안 그 탁상에 집중적으로 관리되는 X Windows 응용프로그람들을 풀어 놓을수 있다.

운영비가 눅은것으로 하여 Linux 는 Windows 95/98/Me 혹은 NT/2000 과 같이 하드웨어나 쏘프트웨어구입에 비용을 들이지 않고도 GUI 망들을 생성할수 있다. 실례로 Linux 가 실행하는 10 개의 탁상으로 된 망을 전개해 보자. 매 탁상들은 8MB 의 RAM 을 갖춘 486 급콤퓨터들이다. 여기에서 Windows plus Microsoft Office 를 실행하는것이 좀 기이한 현상이지만 매 탁상에서 이것을 실현하자면 그에 필요한 쏘프트웨어비용이 미국화페로 300 딸라이상 지출되여야 한다.

그러나 그러한 낡은 하드웨어준위에서도 Linux 는 유일한 응용프로그람봉사체계로서 응용프로그람을 실행하여 X 말단으로 동작하는 매 탁상에 그 응용프로그람들을 제시해 준다. 이 경우 응용프로그람봉사체계는 96MB 의 RAM 을 가진 펜티움 200MHz 체계이다. Linux 사무용제품들의 개발과 무료공개의 혜택으로 쏘프트웨어비용은 본질상 령으로 떨어 지고 있다.

이와 같은 콤퓨터망은 사실상 사용자에게 Windows 95/98/Me 를 태운 32MB 의 RAM 을 가진 펜티움콤퓨터를 다루는 느낌을 준다. 쏘프트웨어관리, 사용자등록자리, 여벌만들기, 체계유지에 대한 모든 처리는 한개 혹은 두개의 봉사기에서 집중적으로 진행된다.

물론 이런 형태의 콤퓨터망은 Windows 에 기초한 대중프로그람을 리용하지 않는다는 자기의 기본전략을 가지고 있다. 그러나 원가가 기본이고 현대적인 하드웨어를 실현시킬 가능성이 제한된 곳에서 Linux 는 조직체범위의 콤퓨터가동환경에 대한 유력한 후보자로 된다.

경로기로서의 Linux

경로기에 대한 개념은 단순하다. 경로기는 둘이상의 콤퓨터망들을 이어 주고 그 망들사이에 통신을 보장하는 장치이다.

경로기는 Windows NT/2000 봉사체계들에서 실행하고 있는 쏘프트웨어경로기로부터 Cisco 와 같이 잘 알려 진 판매업체들에서 제공하는 하드웨어경로기들에 이르기까지 외형과 크기에서 여러가지 형태가 있다. 경로기는 모뎀으로부터 Ethernet 기판, ISDN 접속에 이르는 범위의 기술을 리용하여 다양한 방법으로 망들과 결합할수 있다.

DSL 혹은 케블모뎀과 같은 고속련결방법으로 인터네트와 련결되는 소규모의 콤퓨터망의 사무작업을 실례를 들어 고찰해 보자. 즉 어떤 작은 기업이 인터네트와 하나의 고속회선으로 련결되여 있고 그 회선을 공유하는 둘이상의 콤퓨터들로 구성된 한개의 망을 가지고 있다고 하자.

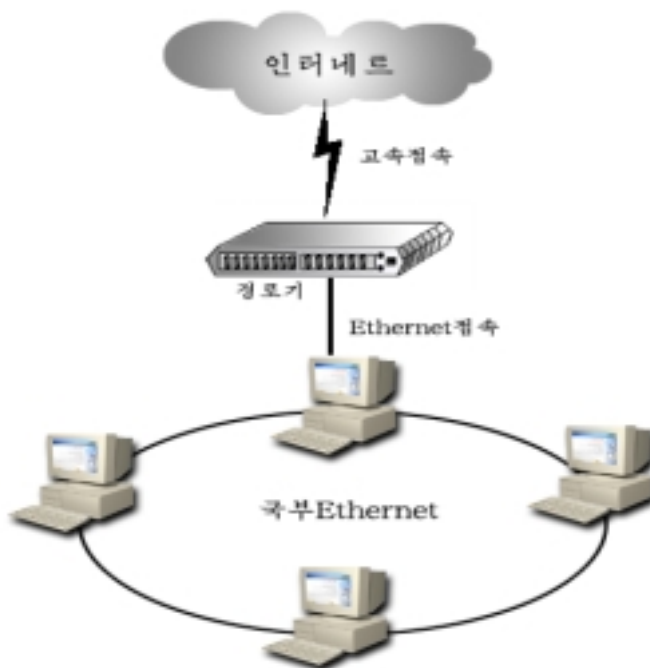



그림 26-1. 대표적인 Ethernet-인터네트경로

이 상태에서 경로기는 이 국부망에 단일한 Ethernet 모선으로 접속되여 있고 인터네트에는 고속회선으로 련결되여 있다. 망에 있는 모든 콤퓨터들이 경로기를 통하여 파케트들을 모두 전송한다. 경로기는 파케트가 국부망의 다른 콤퓨터에 보내자는것인지 외부(Ethernet 의 임의의 곳)로 나가는것인지를 결정한다. 그림 26-1 에는 경로기를 리용한 망의 실례를 그림으로 보여 준다.

그림에서 볼수 있는바와 같이 Ethernet 모선으로 련결된 경로기를 통하여 모든 국부적인 파케트들은 국부망의 임의의 곳까지 전송된다. 만일 파케트가 외부세계에로 전송되도록 예정되여 있다면 고속련결회선을 통하여 밖으로 전송된다. 경로기는 항상 련결되여 있으므로 망파케트들은 인터네트에 련결되기 위하여 모뎀련결을 기다리지 않아도 된다.

그림 26-2 에서는 두개이상의 Ethernet 망과 접속된 경로기를 통하여 두개의 서로 다른 망들이 련결되는 상태를 보여 준다. Fast Ethernet(100Mbps), Gigabit Ethernet (1000Mbps) 혹은 Asynchronous Transfer Mode(155Mbps 이상)와 같은 고속망들의 기본적인 구성도 이와 같다.

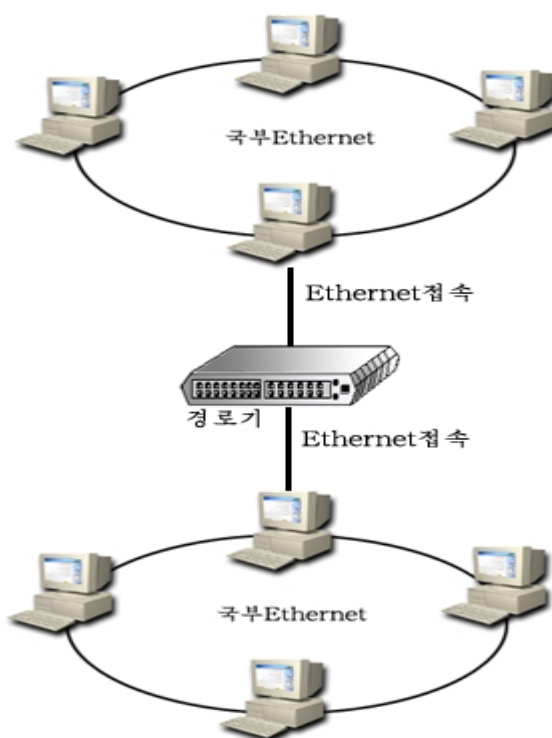



그림 26-2. 경로기를 통한 두개의 Ethernet 망련결

Linux 는 경로기로서 동작하는데 필요한 모든 특징들을 다 가지고 있다.

- 다중 Ethernet 기판들을 지원한다.
- DSL 과 케블모뎀과 같은 규격화된 모뎀과 고속련결을 지원한다 .
- IP 주소의 전송을 지원한다(이것은 Linux 에 기초한 경로기를 통하여 파케트가 수신되고 분석되며 다시 경로화된다는것을 의미한다.)

사실 작은 규모의 Ethernet 호상간 혹은 Ethernet 와 고속인터네트에 접속할 때

Linux 는 전용하드웨어경로기보다 원가가 훨씬 눅으며 이러한 방식으로 구성되면 매우 안전하고 믿음직한 경로기로 될수 있다.

제 31 장 《보안과 값 눅은 경로기서의 Red Hat Linux 7.1》에서 경로기로서 Linux 를 어떻게 사용하는가를 보게 된다.

워크스테이션으로서의 Linux

소규모의 사무작업 환경에서 Linux 의 리용은 실제적인 탁상조작체계를 사용하는것과 같다. Linux 의 응용프로그람이 사무작업에 적합하다면 Linux 탁상환경은 다음과 같은 우점을 가진다.

- 견고성과 성능
- 콤퓨터망관리기에 의한 원격관리
- 망자료보관능력
- Windows 체계에 비한 소유권의 저가격

강력한 Linux 워크스테이션들은 값 비싼 Windows 체계들과 속도를 경쟁하며 Linux 는 Windows 95/98/Me 에서 실행할수 있는 많은 체계들보다도 안전하게 동작하며 보다 든든한 동위망구성을 제공한다.

또한 이미 앞에서 언급한 바와 같이 Linux 는 오래전에 나온 Windows 프로그람도 실행할수 없는 낮은 기종의 콤퓨터들과 오랜 콤퓨터들을 펜티움과 같은 느낌을 가지고 X 말단으로써 유연하게 동작시킬수 있다.

▶미리 알아두기

이 장에서는 Linux 를 가정이나 콤퓨터망사무처리환경에서 리용할수 있는 여러가지 방법들을 보았다.

다음의 장들에서는 보다 더 상세히 이 내용들을 보게 된다. 우선 Linux 안에서 Ethernet 망설치를 어떻게 하는가를 배우게 될것이다. 그때 파일봉사기, 인쇄기봉사기로서의 Linux 에 대하여 배울것이다. 그다음에 Linux 상에서 경로기설정과 인트라네트봉사기설정을 어떻게 하는가를 배우게 될것이다. 마지막으로는 소규모망에서 Linux 를 리용하는데 도움을 줄수 있는 체계관리기술에 대하여 알게 될것이다.

제 27 장 《SOHO 를 위한 Red Hat Linux 7.1 설치》에서는 망환경에서 Linux 를 설치하는 기술을 보게 된다.

제 27 장. SOHO 를 위한 Red Hat Linux 7.1 설치

이 장에서는 제한된 홈워크스테이션에서 Linux 에 적용하는 기본원리를 고찰하고 국부망（LAN）에서 탁상 혹은 봉사기조작체계와 같은 Linux 의 일반적인 역할에 대하여 본다.

Linux 는 실지로 소규모사무, 가정사무(SOHO)에서 점점 더 중요한 역할을 놀고 있다. Linux 는 상업용 Unix 체계에 드는 극히 일부분의 비용을 가지고도 기업급의 콤퓨터망능력을 보장할수 있다.

Linux 는 인트라네트에 필요한 모든 구성요소 즉 Web 봉사, 우편봉사, 새소식봉사들을 포함하여 넓은 범위의 프로그람도구들을 지원하며 대부분의 국부망에서 작업할수 있는 환경을 아주 쉽게 구성한다.

LAN 을 설치하기 위한 프로그람묶음선택

콤퓨터망의 중요한 문제는 Linux 망체계에 어떤 쏘프트웨어들을 설치하겠는가 하는것이다.

제 4 장 《Red Hat Linux 7.1 의 설치》에서 본바와 같이 체계에 콤퓨터망에 설치할 프로그람묶음을 선택할 때 모든 Linux 의 구성요소들을 설치할수 없다면 표 27-1 에 제시된 지도서를 참고한다.

표 27-1. 각이한 콤퓨터망설치를 위하여 제시된 프로그람

체 계 의 형태	설 치 할 프 로 그 람
망에 접속된 워크스테이션	Mail/WWW/News 도구들 망에 접속된 워크스테이션 NFS 봉사기(망에 Unix/Linux 콤퓨터를 접속할 때) SMB(Samba)접속(망에 Windows 콤퓨터를 접속할때)
망관리워크스테이션	모든 망접속워크스테이션에 대하여 다음의것들을 추가한다. IPX/NetWare 접속(만일 Novell 망 혹은 Novell 통신규약을 리용하는 망에 접속되여 있다면) 망관리워크스테이션
망봉사기	모든 망관리워크스테이션에 대하여 다음의것들을 추가한다. 일반 (무명) FTP 봉사기 인쇄봉사기(콤퓨터에 인쇄기가 접속되여 있음을 전제로 하여) 새소식봉사기(망에 새소식그룹양식을 설정하려 할 때) Web 봉사기(이 콤퓨터에서 Web 페지들을 보려고 할 때) DNS 이름봉사기(10 대이상의 콤퓨터망을 가지고 있다고 할 때) PostgreSQL 봉사기(망상에 자료기지지원이 필요할 때)

매 체계에서 목록화된 프로그람묶음들은 다음과 같은것을 제시한다. 실례로 국부망에 Unix/Linux콤퓨터들만이 존재한다면 NFS봉사기프로그람묶음을 설치할 필요는 없다. 특별한 프로그람묶음에서는 사실 보안상 약한 고리로 될수 있다. 실례로 NFS를 리용하지 않고 그것을 설치한다면 콤퓨터망밖에서 누구든지 콤퓨터망에 불법침입하기 위한 수단으로서 NFS를 사용할수 있다.

설치시 망지원의 구성

Linux 설치과정에 Linux 체계에 LAN 지원체계를 구성할수 있다. 제 28 장 《이써네트망에서의 Red Hat Linux 7.1 구성》에서 LAN 에서의 Linux 통신체계를 어떻게 구성하는가를 론의할 때도 구성할수 있다.

LAN 구성작업을 차례로 본다면 우선 Linux 를 설치하기전에 콤퓨터에 이미 망기판이 설치되여 있어야 한다. 이것은 설치프로그람이 망기판을 탐색하기때문에 중요하다.

어떤 기억매체(CD-ROM, 하드디스크구역, 망설치)를 리용해서 체계를 설치하든 상관없이 망환경을 구성할수 있다. 보통 Linux 파일의 설치에 따르는 망구성을 위한 선택항목이 주어 진다. 제 4 장에서는 LAN 지원을 구성하는 선택항목들은 취급하지 않았다. Red Hat Linux 7.1 에서 체계를 설치할 때 콤퓨터를 LAN 에 접속시킬수 있다. 설치프로그람이 망기판을 발견하면 그림 27-1 에서 보여 준 화면이 펼쳐 지게 된다.

만일 이 화면이 나타나지 않으면 이것은 물리적으로 망기판이 설치되여 있지 않다든가 혹은 Linux 가 망기판을 탐색하지 못해서이다. 이 화면이 나타나면 제 28 장에서 Linux 콤퓨터망구성에 필요한 지령을 참고한다.

그림 27-1 은 세개의 부분으로 나누어 져 있다. 이 세개의 부분들은 Linux 망에서 콤퓨터들을 구성하기 위한 기본단계이다. 그러면 이제부터 매 부분들을 구체적으로 보기로 하자.

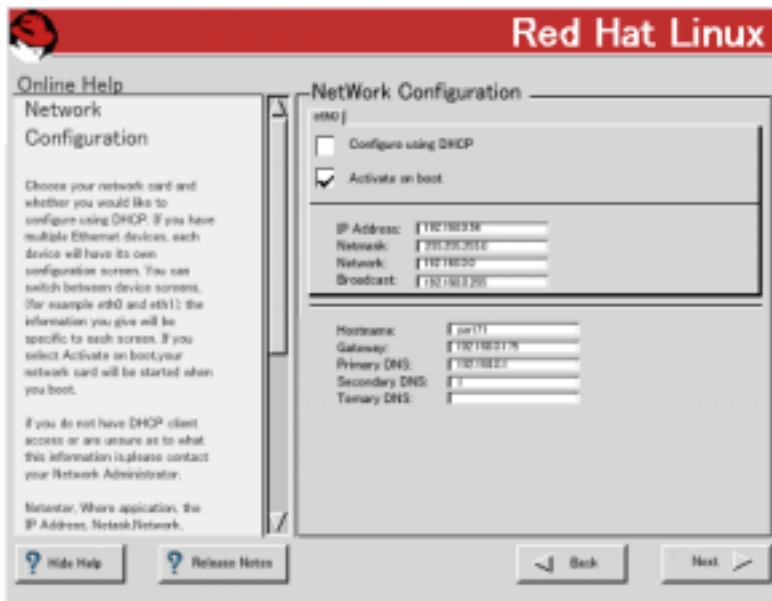



그림 27-1. Linux 설치 때의 망구성

주의 콤퓨터망구성과정에 제기되는 많은 문제들은 제 21 장 《Linux 망설치에 대한 리해》와 제 28 장 《이써네트망에 의한 Red Hat Linux 7.1 의 구성》에서 취급한다. 이러한 론점들은 IP 주소와 망마스크, 이름봉사기와 같은 항목들에 대한 정의를 포함하고 있다.

그림 27-1 은 세개의 부분으로 나누어 져 있다. 이 세개의 부분들은 Linux 망에서 콤퓨터들을 구성하기 위한 기본단계이다. 그러면 이제부터 매 부분들을 구체적으로 보기로 하자.

콤퓨터망기판선택

Red Hat Linux 7.1 을 포함하는 최신배포물들에는 많은 망기판들이 있다. Linux 가 망기판을 발견하면 망기판의 이름을 할당하는데 이 경우에 망기판이름은 eth0 로서 그림 27-1 에서는 하나의 망기판이라는것을 보여 준다. 하나이상의 망기판이 콤퓨터에 설치됐다면 일반적으로 그림 27-1 에서 보여 준것처럼 eth1, eth2 등의 이름으로 표쪽들이 현시된다.

기초콤퓨터망구성

망구성프로그람을 설정하면 두가지 기본선택권을 가지게 된다. 즉 모든 설정을 수동적으로 하거나 매 설정을 할당하는 봉사기에서 자동적으로 콤퓨터를 지적할수 있다.

만일 콤퓨터망이 이미 DHCP 봉사기에 접속되여 있다면 자동적인 구성은 더 쉽다. 그림 27-1 의 모든 망구성요소들을 선택할 필요가 있다면 Configure Using DHCP 을 선택하면 된다. Linux 는 선택한 망기판과 관문의 IP 주소와 함께 자체의 IP 주소를 포함하여 필요한 모든 망설정, DHCP 를 통한 DNS 봉사기를 할당한다.

만일 DHCP 봉사기가 있는지 잘 모른다면 다음절에서 소개하는 수동적인 방법으로 IP 주소를 설정하여야 한다.

주의 일부 Linux 배포물들은 BOOTP 를 리용하여 망을 구성할수 있는 선택항목을 가지고 있다. 이 통신규약은 콤퓨터망의 외부에 있는 DHCP 봉사기가 콤퓨터에 망파라메터를 할당하도록 한다.

수동적인 콤퓨터망주소의 구성

수동적인 콤퓨터망구성은 두개의 부분으로 나누어 본다. 우선 망에서 콤퓨터와 콤퓨터들사이에 기초적인 통신정보들을 설정하여야 한다. 그래야 콤퓨터와 외부망사이의 통신을 설정할수 있다.

수동적인 콤퓨터망구성의 첫 부분은 IP 주소설정이다. 그러나 IP 주소 하나만 가지고서는 콤퓨터망구성을 충분히 할수 없다. 그러므로 이외에 외부망콤퓨터에서 망을 식별하는 망주소, 망마스크들도 설정해야 한다. 망에서 콤퓨터가 다른 콤퓨터들을 식별하게 하려면 방송망 IP 주소가 필요하다. 그림 27-1 에 보여 준것처럼 여기에는 다음과 같은것들이 포함된다.

IP 주소 IP 주소는 192.168.0.223 과 같이 점으로 구분되는 0 부터 255 사이의 수들의 4 개 묶음으로 이루어 진다.

망마스크(Network mask) 망마스크는 대체로 255.255.255.0 처럼 255 들을 포함하는 특수한 IP 주소이다(그러나 늘 그렇지는 않다.).

망주소(address) 망주소는 외부와 접속된 콤퓨터망에서 어떻게 콤퓨터들과 망을 구별하는가를 보여 준다. 콤퓨터망주소가 IP 주소와 망마스크로부터 결정되면 Linux 설치프로그람은 그 주소를 입구점에 자동적으로 써놓는다.

방송망주소(Broadcast address) 이것은 주어 진 콤퓨터망의 모든 콤퓨터들에 통보문을 보낼 때 리용하는 IP 주소이다. 망주소와 같은 방송망주소는 IP 주소와 망주소로부터 결정할수 있으며 Linux 설치프로그람은 그 주소를 자동적으로 어떤 입구점으로 써놓는다.

만일 이 입구점들에 대한 정확한 값을 알수 없다면 망관리기 혹은 ISP 에게 방조를 받아 정확한 값을 취해야 한다. 이미 일부 할당하였던 IP 주소는 리용하지 말아야 한다. IP 주소화에 대한 보다 정확한 리해를 가지려면 IP 주소화와 특히는 IP 판본 4(IPv4)를 론의한 제 21 장을 참고한다.

수동적인 콤퓨터망구성

인터네트를 포함하여 다른 어떤 콤퓨터망과 LAN 의 접속에 대하여서는 기정관문과 주요이름봉사기가 필수적이다. 일반적으로 독립적인 콤퓨터망들에서는 관문이나 이름봉사기들이 필요 없기때문에 이 설정마당들은 공백으로 놔둔다.

두번째 망설치부분에는 그림 27-1 에 기초한 다음과 같은 항목들이 포함되여 있다.

주콤퓨터이름(Hostname) 주콤퓨터이름은 콤퓨터에 권한을 할당해 주는 사용자이름이다. 실례로 mommabears.com 콤퓨터망을 설정했다면 망의 콤퓨터들은 linux1.mommabears.com 와 windows1.mommabears.com 와 같은 주콤퓨터이름을 가진다. 만일 여러 사용자들을 포함하는 령역을 구성해 주지 않는다면 체계설치기간에 주콤퓨터이름은 linux1 혹은 windows1 로 될것이다.

주의 여러 Linux 배포물들은 설치시 령역이름을 설정할것을 요구한다. 대표적인 령역이름들은 Sybex.com, linux.net 와 mommabears.com 파일들에 들어 있다. 령역이 없으면 표준인터네트령역이름은 리용하지 말아야 한다.

Gateway 관문의 IP 주소는 서로 다른 망에 접속된 콤퓨터망의 콤퓨터주소이다. 망이 다른 어떤 망 (인터네트를 포함하면서) 과 접속하지 않는다면 관문주소는 필요 없다.

1 급 DNS(Primary DNS) 령역이름봉사(DNS:A Domain Name Service)로 불리우는 DNS 봉사기는 192.168.55.33.과 같은 IP 주소에 mommabears.com 과 같은 령역이름에 적합한 자료기지를 포함한다. 콤퓨터망에 DNS 봉사기가 있다면 여기에 자체 IP 주소를 입력한다. 그리고 망이 인터네트에 접속되면 여기에 자기 ISP 의 DNS 주소를 입력한다.

2 급 및 3 급 DNS(Secondary and Ternary DNS) 인터네트를 포함하는 다른 망들과 접속한 많은 망들과 체계들은 자체의 령역이름관리체계의 믿음성을 높이기 위하여 하나이상의 이름봉사기들을 리용한다. 사용자나 사용자의 ISP 가 여러개의 DNS 봉사기들을 리용한다면 이 두 마당들에 자체의 IP 주소를 입력한다.

주의 흔히 Linux 체계에서 DNS 봉사기들은 이름봉사기로 알려 져 있다.

이것은 Linux 설치시 콤퓨터망에서 콤퓨터를 구성하는데 필요한 정보이다. 그러나 다음절에서 보게 되겠지만 망에서 Linux 설치파일을 호출한다면 사정은 좀 다르다.

망원천에 의한 설치

만일 설치과정에 체계가 LAN 에 접속한다면 Linux 를 망봉사기에 있는 원천프로그람으로부터 설치할수 있다. 즉 NFS 망과 FTP 봉사기, 지어 Web 봉사기(HTTP)를 통하여 Linux 를 설치할수 있다.

콤퓨터망원천으로부터 체계를 설치하기 위해서는 제 4 장에서 리용한 방법보다 여러 시동디스크를 리용하는것이 더 필요할것이다. 시동디스크는 bootnet.img 파일로부터 만들수 있는데 이 파일은 Red Hat Linux 7.1 설치 CD-ROM 의 images 등록부에 들어 있다. Linux CD-ROM 로부터 만든 시동디스크에 대하여 론의한 제 3 장 《Linux 설치준비》를 참고한다. 이 bootnet.img 디스크로부터 콤퓨터를 시동할 때 각이한 설치화면들이 나타난다. 실례로 Red Hat Linux 7.1 bootnet.img 디스크를 리용하고 설치재촉문들을 거쳐서야 비로소 그림 27-2 에서 보여 준 화면들이 나타난다.

일부 Linux 배포물들에는 설치 CD-ROM 들과 류사한 디스크영상파일들이 있다. 어떤 Linux 배포물들은 이런 형태의 시동디스크조차도 요구하지 않는다. 즉 CD-ROM 들로부터 설치할 때 망설치는 표준선택항목으로 한다.

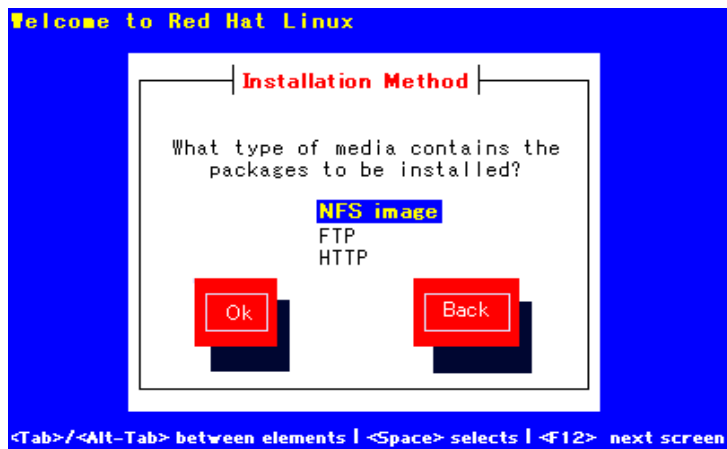



그림 27-2. 망원천으로부터 설치

NFS 원천에 의한 설치

Unix 혹은 Linux 체계콤퓨터로 구성된 망에서 대부분의 영상파일들은 Unix 와 Linux 조작체계에서 자원공유의 표준으로 되고 있는 NFS(Network File System)봉사기에 공동으로 존재한다.

만일 콤퓨터망에서 NFS 봉사기로부터 Linux 설치 CD-ROM 파일들을 호출할수 있다면 이 원천프로그람으로부터 체계를 설치하기 위한 망시동설치디스크를 리용할수 있다.

NFS 봉사기로부터 체계를 설치하기 위해서는 이미 론의된적이 있는 망주소를 설정해야 한다. 콤퓨터망에 DHCP 봉사기가 있다면 이것을 자동적으로 할수 있다.

일단 콤퓨터가 망으로 구성되면 요구하는 Linux 설치파일들을 가지고 있는 NFS 봉사기들을 확인할 필요가 있다. 적당한 Red Hat Linux 7.1 화면을 그림 27-3 에서 보여 주었다. 화면에는 두개의 정보마당이 있다. 즉 리용하려는 NFS 봉사기(혹은 주콤퓨터이름)의

IP 주소와 Red Hat Linux 7.1 설치원천파일들이 들어 있는 봉사기의 경로이다.

실례로 Red Hat Linux 7.1 설치파일들이 /mnt/cdrom 에서 NFS 봉사기에 설치한 원격 CD-ROM 에 존재한다고 가정하자. 또한 NFS 봉사기가 IP 주소 10.10.10.1 을 가진다고 가정하자. 이 경우에 NFS 봉사기이름으로는 **10.10.10.1** 을 입력하고 Red Hat 등록부로는 **/mnt/cdrom** 을 입력한다.

설치수속의 나머지단계는 국부 CD-ROM 구동기로부터 설치할 때와 같은 단계로 설치한다.

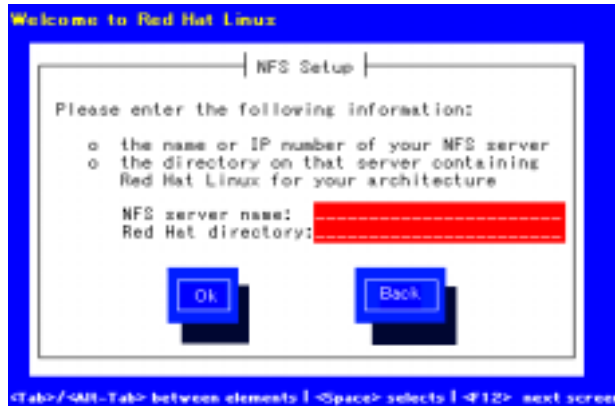



그림 27-3. NFS 봉사기로부터 설치

FTP 원천에 의한 설치

한편 FTP 봉사기로부터 Linux 를 설치할수 있다. 이 봉사기는 국부망에 있을수 있거나 그렇지 않으면 인터네트상에서 Linux 설치파일들을 가지고 있는 많은 FTP 봉사기들중의 어느 하나에 있을수 있다.

주의 고속콤퓨터망 또는 하기 쉬운 인터네트망접속이 없다면 인터네트 FTP 봉사기로부터 체계를 설치하는것은 불가능하다. 만일 설치형태를 이렇게 정한다면 그것은 며칠 걸릴것이다. 설치하던 도중에 인터네트접속이 중단되면 처음부터 다시 시작해야 한다.

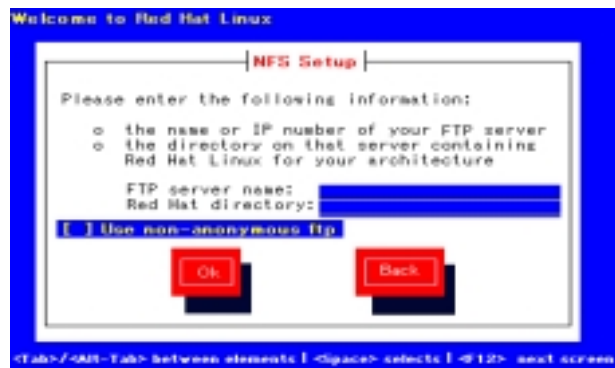



그림 27-4. FTP 봉사기에 의한 설치

만약 FTP 선택항목을 리용하려 한다면 그림 27-2 에서 보여 준 설치화면에서 FTP 를 선택하여야 한다. 그다음 이미 론의한 망주소를 설정해야 한다. 망에 DHCP 봉사기가 있다면 이것은 자동적으로 수행된다.

그러면 FTP Setup 창문이 그림 27-4 에 보여 준것처럼 나타난다. 이 창문에서 다음과 같은 3 개의 정보들을 받을수 있다.

FTP Site Name 이것은 리용하려고 하는 FTP 봉사기이름이다. 실례로 Red Hat FTP 봉사기로 설정하려고 한다면 **ftp.redhat.com** 을 입력한다.

Red Hat Directory 이것은 Linux 배포물들의 원천파일이 들어 있는 FTP 봉사기의 등록부이다. 실례로 선택된 FTP 봉사기가 Red Hat 7.1 설치파일들을 등록부 /pub/mirrors/redhat/i386/RedHat 에서 취한다면 **/pub/mirrors/redhat/ i386** 을 리용한다(“RedHat”라는 마지막단어생략을 주의해야 한다.).

Use Non-Anonymous FTP 만일 사용하려는 FTP 봉사기에 망가입하려면 이것을 선택한다. 이것을 선택하면 닉명이 아닌 FTP 봉사기가입에 리용하는 등록자리이름과 암호문이 필요하다.

Web 봉사기에 의한 설치

일부 대용봉사기들은 FTP 를 호출하지 않고 HTTP 를 리용하여 Web 봉사기들을 호출한다. 일부 사용자들은 Web 봉사기로부터 Linux 설치에 대해서도 정보의 내리적재를 더 쉽게 한다.

주의 고속콤퓨터망 혹은 하기 쉬운 인터네트망접속이 없다면 인터네트 FTP 봉사기로부터 체계를 설치하는것은 불가능하다. 만일 설치형태를 이렇게 정한다면 그것은 며칠 걸릴것이다. 설치하던 도중에 인터네트접속이 중단되면 처음부터 다시 시작해야 한다.

만일 HTTP 선택항목을 리용하려 한다면 그림 27-2 에 보여 준 설치화면에서 HTTP 를 선택하여야 한다. 그러면 이미 론의한것처럼 망주소를 설정해야 한다. 콤퓨터망에 DHCP 봉사기가 있다면 이것은 자동적으로 수행된다.

그러면 그림 27-5 에서 보여 준바와 같이 HTTP Setup 창문이 나타난다. 이 창문에서 Linux 설치파일과 적중한 등록부를 내리적재하는 곳에서 Web 싸이트를 받을수 있다.

그림 27-5. Web 봉사기로부터 설치

실례로 Red Hat Linux 7.1 설치파일들이 HTTP 봉사기의 등록부 pub/mirrors/redhat/i386/RedHat 에 설치되여 있다고 하자. 또한 HTTP 봉사기가 **www.redhat.com**에 있다고 가정하자. 그때 Web Site Name 으로는 **www.redhat.com** 를 입력하고 Red Hat Directory 로는 **/pub/mirrors/redhat/i386** 을 입력한다.

설치수속의 나머지단계는 국부 CD-ROM 구동기로부터 체계를 설치할 때와 같은 단계로 처리한다.

▶미리 알아두기

지금까지 Linux 망체계를 설치했다면 이제부터는 Linux Ethernet 망구성에 대하여 구체적으로 보려고 한다. 이 정보는 Ethernet 국부망 (LAN) 에서 망으로 되여 있지 않는 Linux 체계를 설치하거나 또는 필요한 망구성정보들을 재구성하도록 한다.

일단 이에 대한 상세한 정보를 배웠다면 Linux 가 다음과 같은것을 포함하는 소규모인트라네트망에 적용하는 일부 규칙들을 보여 주는 책의 나머지부분에 시간을 소비할것이다.

- 값 눅은 경로기 혹은 방화벽으로 봉사하기
- 인트라네트 Web 봉사기로서 동작하기
- 망의 전자우편봉사를 조종하기

제 28 장. 이써네트망을 위한 Red Hat Linux 7.1 의 구성

제 27 장 《SOHO 을 위한 Red Hat Linux 7.1 설치》에서 콤퓨터에 어떻게 Linux 를 설치하며 처리시 그것을 어떻게 TCP/IP 이써네트망에 련결하는가를 학습하였다. 또한 Linux 설치중에 망기판을 구성하는 기초적인 단계들을 학습하였다.

이 장에서는 이미 설치된 Linux 체계를 어떻게 Linux 망에 접속시키는가를 배우게 된다. 여기서 망을 위한 개인용콤퓨터의 준비와 망구성, 망검사를 단계적으로 거치게 된다. 이 장에서는 완전한 처리를 취급한다.

주의	일반적으로 IEEE 표준 802.2 와 802.3 으로 알려 진 Ethernet 는 망기술의 넓은 범위에서 다만 가장 보편적인것이다. Linux 를 설치하면 통표고리와 ARCNet, 고속이써네트, Gigabit 이써네트, 비동기전송방식(ATM)을 포함하여 각이한 망기술로 작업을 진행할수 있다. 기술은 여러가지이나 설치기술은 지령행대면부를 리용한다는 점에서 아주 비슷하다. 좀더 구체적인 정보 특히 ATM 뿐 아니라 여러가지 이써네트항목은 http://www.linuxdoc.org/HOWTO /Ethernet -HOWTO-2.html에 직결로 호출가능한 Ethernet-HOWTO 의 두번째 단락에 있다.

이 장은 Linux 2.2.x 핵심부와 련관된 전통적인 망지령들에 기초하고 있는데 이것들은 여전히 Red Hat Linux 7.1 에서 리용된다. Linux 2.4 핵심부와 련관된 일부 새로운 망지령들이 있는데 더 구체적인 정보는 http://www.ds9a.nl/2.4 Networking/HOWTO //cvs/2.4routing/output/2.4networking.html 에서 얻을수 있다.

망준비와 구성

이전의 장에서 본것처럼 망작업환경을 마련하는데는 기본적으로 두 단계가 필요하다.

- 망기판설치와 설정
- TCP/IP 파라메터설정

이 장은 Linux 가 설치된후에 이러한 본질적문제들이 어떻게 구성되는가 하는것을 본다. 이 프로쎄스는 망에서 새로운 PC 을 설치하고 현재의 PC 의 구성을 변화시키는데 리용한다.

망기판설치와 설정

망기판을 설치하고 설정하는것은 전적으로 리용하려는 망의 형태에 의존한다. 제 27 장에서 론의한 망기판들을 고찰하고 리용해야 할 망기판의 형태와 Linux 에서 그것을 알맞게 구성하자면 어떤 정보들이 요구되는가를 결정해야 한다. 기판을 선택하면(여기서는 많은《이름 없는》이써네트기판들에 대해 대체로 기정항목인 Novell 계 2000 계렬기판을 선택한다고 가정한다.) 먼저 그 기판에 따라 나온 사용설명서에 따라 하드웨어를 설치해야 한다. 일부 경우에 콤퓨터를 재기동하고 Red Hat Linux 7.1 이 새로운 구동기를 적재할수 있게 해야 한다. 다른 경우에 Red Hat Linux 7.1 은 자동적으로 기판을 검색하고 구성하며 설치한다. 망기판이 능동인가를 보기 위해 뿌리사용자방식으로 들어 가서 /sbin/ifconfig 지령을 실행한다. 다음과 같은 몇가지 결과가 나타나면 Linux 는 이써네트(eth0)망기판을 콤퓨터에 알맞게 설치하였다는것이다(loopback 장치인 lo 도).

```
eth0    Link encap:Ethernet   HWaddr 00:50:56:85:00:23
        inet addr:192.168.0.135   Bcast:192.168.0.255   Mask:255.255.255.0
        UP BROADCAST RUNNING MULTICAST   MTU:1500   Metric:1
        RX packets:50 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueue1en:100
        Interrupt:9 Base address:0x1000

lo       Link encap:Loca1 Loopback
        inet addr:127.0.0.1   Mask:255.0.0.0
        UP LOOPBACK RUNNING   MTU:16436   Metric:1
        RX packets:6 errors:0 dropped:0 overruns:0 frame:0
        TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
        co11isions:0 txqueue1en:0
```

이것이 하드웨어와 일치하면 이 장의 나머지를 읽고 망기판을 보존하는 도구들을 시험해 볼수 있다.

그러나 망기판을 설치하고 eth0 단락이 보이지 않으면 Linux 는 망기판을 인식하지 못한것이다. 한편 하나이상의 망기판이 있는데 하나의 eth0 단락만이 보인다면 Linux 는 다른 망기판들을 인식하지 못한것이다. 이 장의 나머지부분을 읽고 망기판(들)을 어떻게 구성하는가를 배워야 한다.

주의	여기서는 이써네트망기판리용을 전제로 한다. 다른 형태의 망기판들은 다르게 표시되는데 실례로 콤퓨터상에서 처음의 ARCNet 기판은 arc0 으로 된다.

Linux 가 망기판을 자동검색하지 못한다면 수동적으로 Linux 가 하드웨어를 지원하도록 해야 한다. Linuxconf 를 가동하여 시작한다. xterm 창문에서 지령재촉문으로 다음의 지령을 건입력하여 지령행으로부터 Linuxconf 를 기동할수 있다.

$ linuxconf-auth

뿌리사용자가 아니라면 Red Hat Linux 7.1 은 Linuxconf 를 시작하기전에 뿌리통과암호를 요구한다. Linux 가 이 지령을 인식하지 못하면 아마 15 장에서 보여 준 매 지령들에 Linuxconf 를 설치해야 한다.

일단 Linuxconf 가 기동하면 Networking -> Client Tasks -> Host Name And IP Network Devices 를 선택하여 그림 28-1 에서 보여 준 화면을 얻는다.

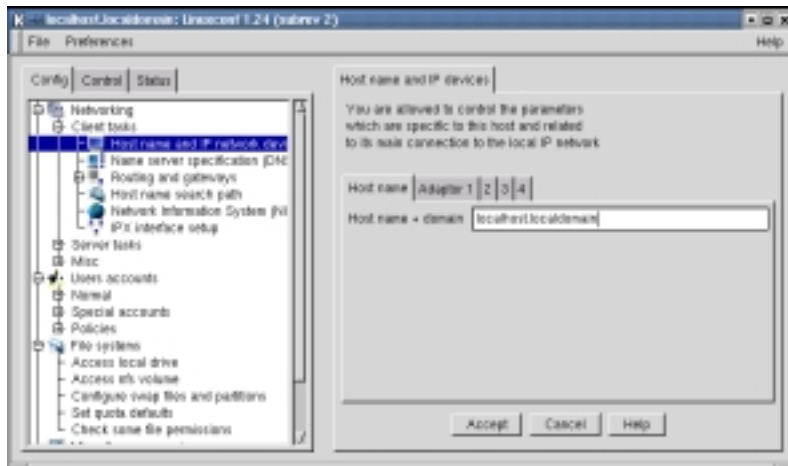



그림 28-1. Linuxconf 지령으로 콤퓨터망구성

Linuxconf 의 이 판본에서 Host Name And Devices 단락은 6 개의 표쪽을 포함한다. Host Name 표쪽은 콤퓨터와 결합된 주콤퓨터이름을 나타낸다. 다른 5 개의 표쪽들은 요구하는 망기판이나 접속기들을 구성하도록 한다.

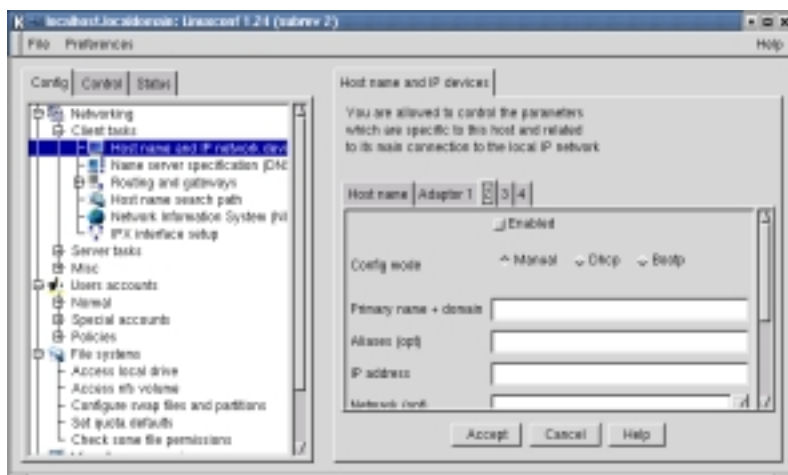



그림 28-2. 새로운 망기판추가

하나의 일반적인 문제는 두개의 망기판들의 검색이다. 두개의 각이한 망에 접속하도록 콤퓨터를 설정할 때 두개의 망기판을 요구한다. Adapter 1 표쪽을 찰칵한다. Linux가 첫번째 망기판을 인식하였다면 해당 구성자료는 보통 이 표쪽에 나타난다. 두번째 기판을 구성하려면 Adapter 2 표쪽을 찰칵한다. 그림 28-2 에서 보여 주는것처럼 그것이 비였다고 가정한다.

보는바와 같이 매개 망기판과 결합된 몇개의 설정항목이 있다. 일부는 망기판구성에서 금지되여 있으며 한편 다른것들은 그렇지 않다.

Enabled 기판을 능동화하려면 이 항목을 선택한다.

Config Mode 망에 DHCP 봉사기를 가지고 있지 않다면 Manual 을 선택한다. 국부망에 DHCP 봉사기를 가지고 있다면 봉사기가 망기판을 구성하도록 하기 위해 Dhcp 를 선택한다. 린접한 망에 DHCP 봉사기를 가지고 있다면 봉사기가 기판을 구성하도록 하기 위해 Bootp 를 선택한다.

Primary Name + Domain 콤퓨터와 령역이름을 입력한다. 실례로 콤퓨터가 이름이 mommabears.com 망의 linux7.1 이라면 여기에 linux 7.1.mommambears.com 을 입력한다. 한편 이 항목을 비여 있는 그대로 남겨 두고 후에 이것을 /etc/hosts 파일에 설정할수 있다.

Aliases 콤퓨터에 다른 이름이 있다면 여기에 그것들을 입력한다. 하나이상의 별명이 필요하면 정확히 그것들을 공백으로 갈라 준다.

IP Address 제 21 장《Linux 망구성의 리해》에서 론의한바와 같이 IP 주소는 점으로 구분된 0 과 255 사이 4 개의 수이다. Config mode 가 Manual 이면 여기에 이 망기판의 IP 주소를 입력한다.

Netmask 망마스크를 보여 준다. 《opt》표식대신에 매개 망기판에 대한 망마스크를 요구한다. 제 21 장에서 론의한바와 같이 망마스크들은 다른 콤퓨터가 어떤 다른 콤퓨터들이 국부망에 접속되는가를 결정하도록 한다. 일부 전형적인 망마스크들을 다시 보자면 내림차림표를 리용할수 있다.

Net Device Linux 가 망기판과 련관된 이름이다. 만약 Ethernet 망이라면 망장치이름들은 대체로 eth0, eth1, eth2 등으로 된다. 이 경우에 두번째 이써네트결합기판(eth1)을 설정하면 상태는 달라 질수 있다.

Kernel Module 이것은 망기판과 결합하려는 구동기이다. 내림차림표에서 보여 준 이 목록은 포괄적이지 못하다. 즉 망기판이 각이한 구동프로그람을 요구하며 그것이 /lib/modules/2.4.2-2/kernel/drivers/net 등록부에 설치되여 있다면(혹은 그 보조등록부중 어느 하나) 여기에 모듈이름을 넣는다. 혹은 이 설정을 공백으로 남기고 후에 설명하는 insmod 지령을 리용할수 있다.

알려두기 망기판과 결합된 핵심부모듈을 인식할수 없다면 http://www.linuxdoc.org/HOWTO/Ethernet-HOWTO.html에서 Ethernet- HOWTO 를 검사한다. 혹은 망기판에 대한 구동프로그람디스크나 Web 홈페지를 참고한다. 일부 망구동프로그람들은 Linux 구동프로그람과 설치지령들을 포함한다.

I/O Port 망기판과 결합된 기억기주소이다. Red Hat Linux 7.1 은 보통 이것을 자동적으로 가리킨다. 다음단계들에서 문제가 생기면 망기판과 결합된 표준 I/O 주소를 찾아서 여기에 그것을 입력한다.

Irq 망기판과 결합된 새치기요구통로이다. 보통 Red Hat Linux 7.1 은 보통 이것을 자동적으로 가리킨다. 다음단계들에서 문제가 생기면 망기판과 결합된 표준 IRQ 통로를 찾아서 여기에 그것을 입력한다.

적당한 본문칸들을 채웠다면 다음단계에서는 그것을 검사한다. 표쪽밑에서 Accept 를 마우스로 찰칵하고 그때 Linuxconf 창문의 왼쪽 웃모서리에 있는 File ⇨ Act/Changes 를 선택한다. 그러면 그림 28-3 에 보여 주는것처럼 Status Of The System 대화칸이 나타나 Linuxconf 는 변화된 정보들을 설치해야 한다는것을 알려 준다. 이 경우에 망데몬을 재시동하고 망구성파일들을 다시 읽어 넣을것을 권고한다. 그것을 실행하기 위하여 Do It 를 찰칵한다.

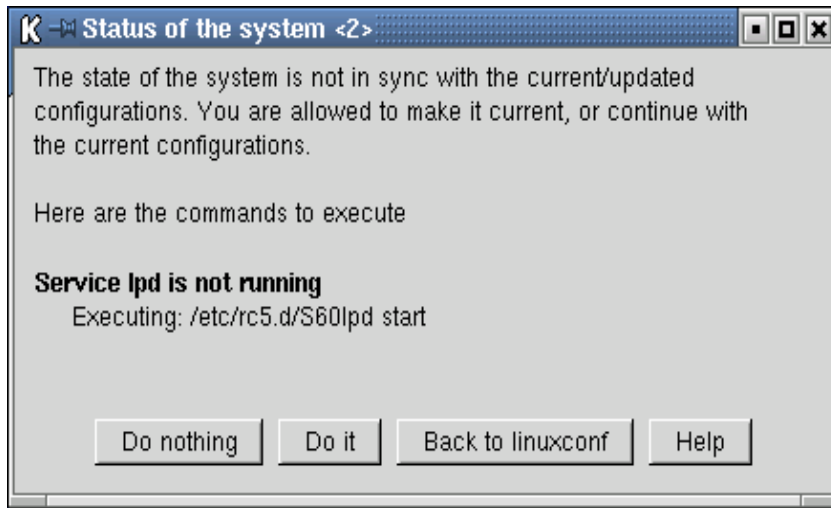



그림 28-3. Linuxconf 지령에 의한 구성정보들의 변화

이미 설명한 /sbin/ifconfig 지령을 다시 실행한다. Linux 가 현재망기판을 인식했다면 망을 유지하는 관점에서 이 장을 읽을수 있다.

모듈의 적재상태검사

만일 Linuxconf 으로 망기판을 설정할수 없다면 다음단계에서 구성을 확인해야 하는데 그것은 쉽지 않다. 다음단락에는 망을 동작시킬 때까지 Ethernet 기판이 정확히 구성되였는가를 알아 보는 빠른 방법이 없다. 그러나 적재가능한 핵심부모듈을 설치하기 위해 insmod 지령을 수동적으로 리용하는 방법으로 모듈이 적재되였는가를 검사할수 있다.

이 작업을 수행하기 위하여서는 조종탁지령행으로 가든가 혹은 뿌리사용자로 xterm 창문을 열어야 한다. 먼저 다음과 같은 지령으로 현재록부를 변화시켜야 한다.

```
$ cd /lib/modules/2.4.2-2/kernel/drivers/net/
```

이 지령은 이 책에 씌여 져 있는 Red Hat Linux 7.1 배포물을 설치하고 핵심부를 갱신하지 않았음을 전제로 한다. 이 경우에 등록부경로에 반영되여 있는 핵심부판본은 2.4.2-2 이다. 일단 등록부가 정확하다면 insmod 지령으로 요구하는 모듈(이 경우에는 Novell 계렬의 망기판)을 적재한다.

$ insmod ne.o

만일 Linux 가 요구한 모듈을 인식하지 못한다면 이 등록부(혹은 자체보조등록부) 안에 있는 유효한 모듈목록을 다시 한번 본다. 만약 확실한 모듈이라면 입력한 문법을 검사하고 다시 실행시킨다. 모듈들의 파일이름이 확장자가 o 이라는것에 주의한다.

성과적으로 구성되였다면 망기판구동프로그람이 적재되고 통보문들이 망구동프로그람에 의해서 발생한다. 그것들은 3c59x 구동프로그람의 실례에서 보여 주는바와 같이 복잡하다.

```
3c59x.c:v0.46C 10/14/97 Donald Becker
    http://cesdis.gsfc.nasa.gov/linux/drivers/vortex.html
loading device 'eth0'...
etho: 3Com 3c905 Boomerang l00baseTx at 0x1440,00:60:08:71:ad:8c, IRQ 111
   8k word-wide  RAM  3:5 Rx:Tx  split,  autoselect/MII  interface.
eth0: MII transceiver found at address 24.
eth0: Overriding PCI latency timer (CFLT)  setting of 64, new value is 248.
```

혹은 다음과 같이 단순하게 할수도 있다.

Using /1ib/modu1es/2.4.2-2/kerne1/drivers/net/pcnet32.o

오유가 없다면 모듈들은 성과적으로 적재되고 망기판은 정확하게 구성된다. 망기판을 구성하기가 힘들면 http://www.linuxdoc.org /HOWTO /Ethernet-HOWTO.html에서 Ethernet-HOWTO 을 참고한다.

TCP/IP 파라메터의 설정

다음단계는 망구성편의프로그람을 리용하여 TCP/IP 파라메터들을 설정하는것이다. 이를 위하여 말단창문을 기동하여 뿌리사용자방식으로 가입하고 다음과 같은 지령을 입력한다.

$ netcfg

이것은 그림 28-4 에 보여 준 Red Hat Linux Network Configurator 를 호출한다.

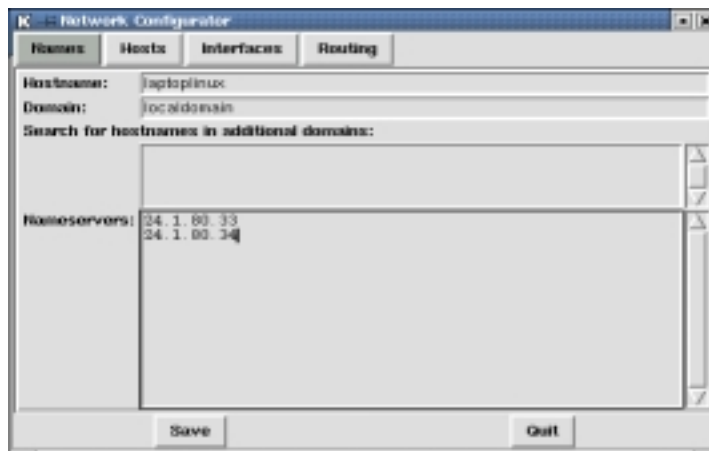



그림 28-4. 망구성도구프로그람

이 편의프로그람에서 접속한 망과 관련된 체계에 대한 명백한 정보를 지정한다. 이 정보는 다음과 같다.

- 이름, 령역, 주소 그리고 콤퓨터의 DNS 봉사기
- 망에 있는 다른 콤퓨터들의 이름과 주소들
- 유용한 망의 매 대면부를 어떻게 리용하는가?
- 목적지에 도착하기 위하여 정확한 통신량로정을 어떻게 정하겠는가?

주의 현재 존재하는 망과 련결되여 있다면 이 부분에 필요한 많은 정보들은 망의 망관리기에게서 찾아 볼수 있다. 만약 인트라네트로부터 망을 전개할 계획이라면 TCP/IP 망화(Networking)와 망설계에 대한 기초지식들을 읽어 보는것이 좋다. 이에 대한 좋은 자원은 Linux 망관리기안내서인데 직접 http://www.linuxdoc.org/LDP/nag2에서 찾아 볼수 있다.

이름설정

첫 단계는 콤퓨터의 이름과 이름봉사기를 구성하는것이다. 이것은 Network Configurator 대화칸을 열었을 때 기정으로 표시되는 Names panel 에서 수행된다. 앞에서 본 그림 28-4 는 이 기정 panel 을 보여 준다.

Names panel 에서 주요한 3 가지 기본정보를 제공해야 한다.

Your hostname 이것은 콤퓨터에 할당된 이름이다. 이 이름은 망상에서 유일하여야 한다.

The domain for your network 실례로 콤퓨터에 완전히 적합한 령역이름이 linux1.Sybex.com 이라면 그 망에 대한 령역이름은 Sybex.com 이다. 령역이름은 특별히 Web 봉사기처럼 인터네트봉사기로서 동작하지 않는 국부망상의 콤퓨터들에는 필요없다.

Your DNSes or nameservers 이것은 한개 행이 망에 령역이름봉사를 제공하는 콤퓨터들의 목록이다. 령역이름봉사는 내부망과 외부망상의 콤퓨터이름들을 콤퓨터들이 다른것과 통신하는데 리용할수 있는 IP 주소로 변환한다.

DNS 들은 IP 주소에 의해서 형식적으로 배렬된다.

주콤퓨터들의 문서처리

다음단계는 콤퓨터에 대한 주콤퓨터표를 취급한다. Hosts 단추를 찰칵한다. 실지로 파일 /etc/hosts 에 보관된 주콤퓨터표는 이름탐색의 가장 단순한 형태이다. 망우에서 /etc/hosts 를 리용한다면 망우의 모든 콤퓨터들은 같은 파일/etc/hosts 로 설정되여야 한다. 소규모의 망우에서는 DNS 를 설정하는것보다 /etc/hosts 파일들을 보존하는것이 덜 위험하다. 국부망을 위한 DNS 가 있다면 모든 /etc/hosts 파일은 두개의 입구점을 요구한다.

- 국부적인 주콤퓨터입구점
- 콤퓨터의 이름과 IP 를 가진 입구점

목록에 입구점을 추가하기 위하여 Add button 을 찰칵한다. 그러면 그림 28-5 와 같은 대화칸이 나타난다.

이 대화칸에는 3 개의 마당이 있다. 최소한 구성하려는 콤퓨터이름과 IP 주소를 (linux 1.mommabears.com 과 같이)입력한다. Nicknames 마당에는 공백으로 분리된 콤퓨터의 대리이름을 입력할수 있다. 보통 체계관리기들은 이 마당에 주콤퓨터이름을 넣는다(linux 1.mommabears.com 인 경우에는 주콤퓨터이름이 linux1 이다.). 끝마칠 때 Done 단추를 찰칵한다.

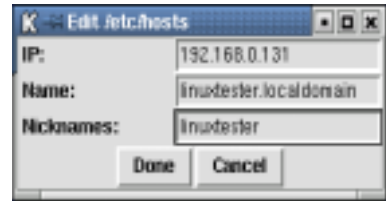



그림 28-5. 주콤퓨터추가

주콤퓨터표에 항목들을 편집하는것은 새로운 입구점들을 추가하는것과 류사하다. 현재 존재하고 있는 입구점들을 선택하고 Edit 단추를 찰칵한다. 그러면 그림 28-5 에서 보여 주는것과 같은 대화칸이 나타나는데 여기에는 주콤퓨터표에 새로운 입구점을 추가할 때와 같은 3 개의 마당이 있다. 일단 입구점내용의 편집이 끝나면 주콤퓨터표의 변화들을 알려 주기 위하여 Done 단추를 찰칵한다. 한가지 가능한 결과를 그림 28-6 에서 보여 준다.

그림 28-6. 전형적인 주콤퓨터표

대면부관리

다음조종판은 체계상에서 망대면부들을 구성하는것이다. Interfaces 단추를 마우스로 찰칵한다. 대면부들은 모뎀에 기초한 접속과 Ethernet 기판을 통한 접속들, 지어는 Token Ring 접속들을 포함할수 있다. 기정으로 이 표는 적어도 lo(이것은 국부적인 재귀순환입구점이고 현재망 없이도 작업할수 있는 Linux 를 위한 표이다.)라고 불리우는 하나의 입구점이 포함된다. 매 대면부는 IP주소와 결합된다.

새로운 대면부를 추가하는 첫 단계는(콤퓨터가 그 곳에 없다고 가정할 때) Add 단추를 찰칵하는것이다. 대화칸은 그림 28-7 에서와 같이 나타나는데 거기서 추가하려고 하는 그림칸대면부의 형태를 선택할수 있다.

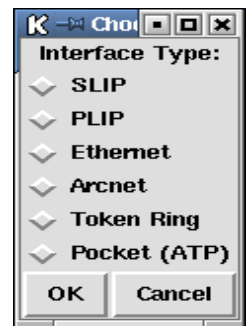



그림 28-7. 대면부 형태의 그림칸

국부망(LAN)인 경우에 망기판에 대한 모듈을 설정할 때 미리 해놓은 같은 형태의 장치를 선택한다. 이것은 Ethernet 대면부일수 있다. OK 단추를 찰칵하면 대화칸이 나타나는데 여기서 대면부속성을 설정할수 있다. Ethernet 대면부인 경우에 대화칸은 그림 28-8 에서와 같이 나타난다.

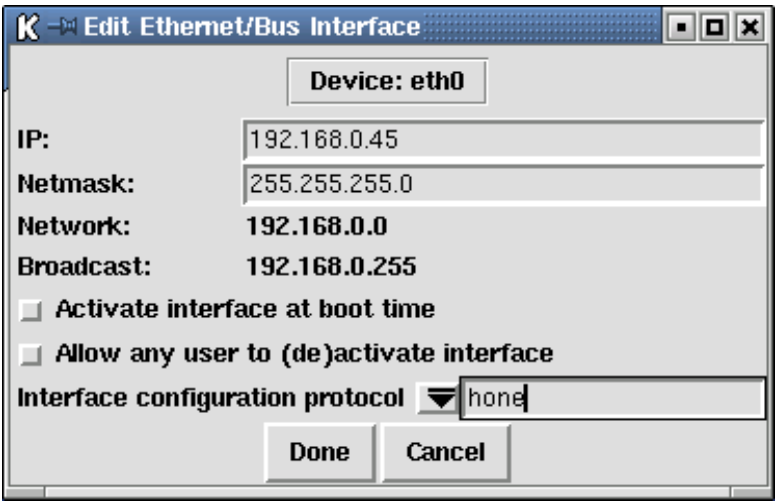



그림 28-8. Ethernet 대면부구성

망이 국부 혹은 원격 DHCP 봉사기(BOOTP 을 거처)를 포함한다면 대면부의 IP 주소와 망마스크를 제공해야 한다(뒤에서 대면부구성규약을 서술한다.). 만일 하나의 망대면부만을 가지고 있다면 IP 주소는 Names 와 Hosts panel 에서 콤퓨터에 입력한것과 같은 것이여야 한다. 다른 가능한 선택들은 다음과 같다.

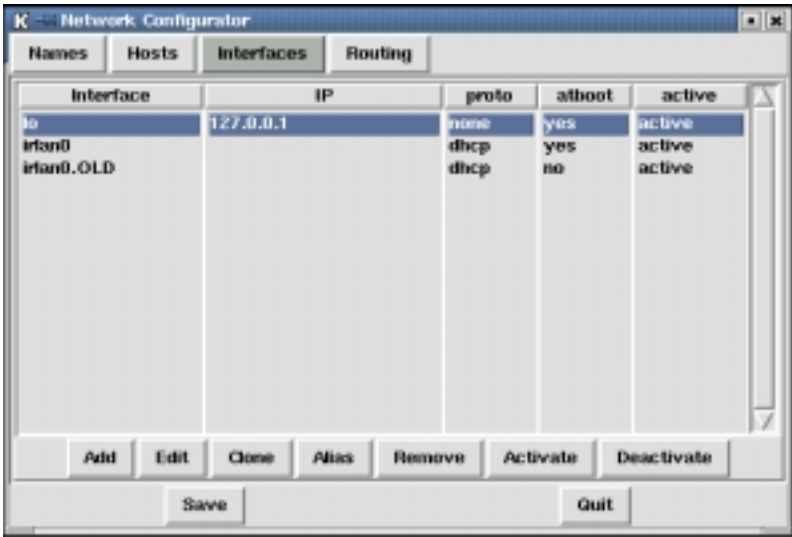



그림 28-9. Ethernet 대면부를 능동으로 만들기

Activate Interface At Boot Time 콤퓨터를 기동할 때마다 망을 쓸수 있게 하려

면 이것을 선택해야 한다. 만일 이것을 선택하지 않으면 수동으로 망을 리용할수 있게 만들어야 한다.

Allow Any User To(De)activate Interface 일반적으로 뿌리사용자만이 망대면부를 능동화하거나 비능동화할수 있다. 임의의 사용자가 이것을 하도록 허가하는것은 사용자가 망접속을 설정하기 위하여 전화를 거는 전화대면부에서 특별히 쓸모가 있다. 기동시에 이 대면부는 자동적으로 능동상태로 되기때문에 Ethernent 국부망련결을 정확히 취급하고 대면부를 활성화한다면 이 속성을 가능하게 만들 필요가 없을것이다.

Interface Configuration Protocol 망이 국부 혹은 원격(BOOTP 를 거쳐)DHCP 봉사기를 가지고 있다면 콤퓨터에 대한 IP 주소 혹은 망마스크를 입력할 필요가 없다. 이러한 DHCP 봉사기들중에서 하나도 가지고 있지 않다면 None 을 선택한다. 그렇지 않으면 망에 있는 DHCP 봉사기의 형태를 선택한다.

일단 대면부를 설정해 놓으면 Done 을 마우스로 찰칵한다. 그림 28-9 에서 보여 준 대면부화면에서 직접 그것을 선택하 고 Activate 단추를 마우스로 찰칵하여 대면부를 능동화할수 있다.

경로기설정

어려운 망구성의 마지막단계는 경로기설정이다. 그림 28-10 에서 보여 준 경로조종판을 현시한 창문의 꼭대기에서 Routing 단추를 찰칵한다. 경로기표는 미리 정해 놓은 자료를 각이한 망에 어떻게 보내는가를 콤퓨터에 알려 주는데 리용된다.

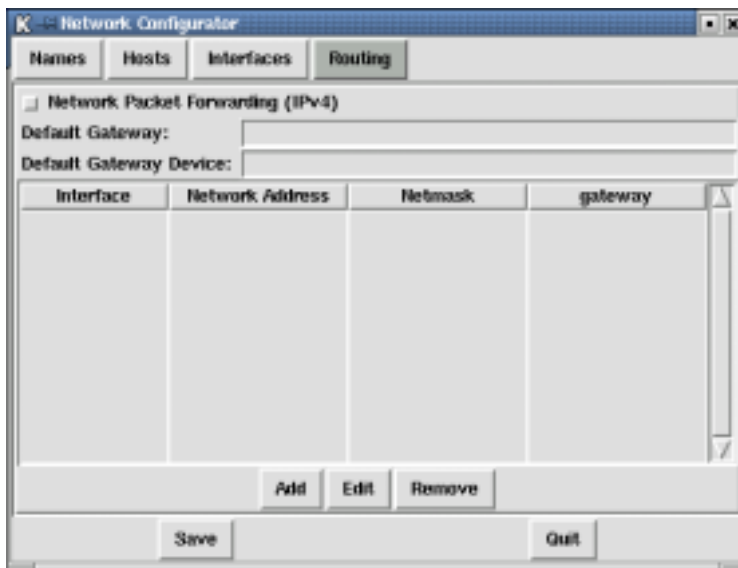



그림 28-10. 경로기조종판

망이 하나의 기정관문콤퓨터를 가지고 있다면 기정관문콤퓨터의 IP 주소를 입력한다. 콤퓨터가 하나이상의 망기판을 가지고 있다면 Default Gateway Device 로서(실례로 eth0 혹은 ppp0) 직접 관문(실례로 eth0 혹은 ppp0)에 접속되는것을 선택한다. 다른 장치들은 국부망에서 정적경로기로 된다.

실례로 콤퓨터가 두개의 이써네트기판을 가지고 있으며 그중 하나는 기본단체망 (eth0)에 접속되여 있고 다른 하나는 일부 프로그람들을 검사하는데 리용되는 소규모국부망 (eth1)에 접속되여 있다고 가정한다. 그림 28-11에서 보여 준 Edit Static Route 대화칸을 열기 위하여 Add 를 찰칵한다.이 panel 에서 내부의 정적경로기를 설정하고 정적경로기를 eth0 으로 할당하였는가를 확인하고 Done 을 찰칵한다.

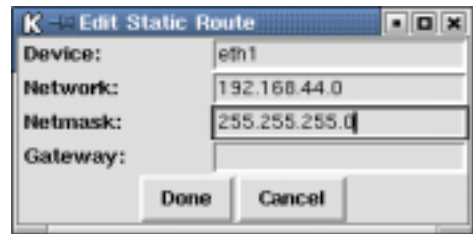



그림 28-11. 국부정적경로기설정

경로기구성은 그림 28-12에서 보여 주는것과 같이 eth0에서 기정관문을, eth1에서 시험망을 위한 정적경로기를 포함한다. 접속하려는 국부적인 Ethernet 망을 위하여 정적경로기를 포함하는것은 일반적이다.

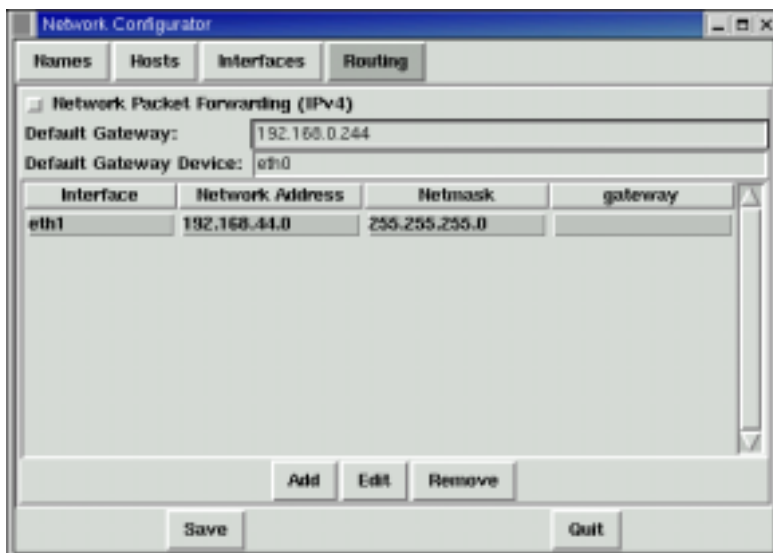



그림 28-12. 다중경로기

콤퓨터망시험

다음단계에서는 망접속을 검사하여 모든것이 정상인가를 알아 본다. 먼저 기본적인 망의 접속상태를 검사해야 한다. 이것을 ping 지령을 리용하여 진행할수 있는데 이 지령은 정해 진 IP 주소에 조회를 보내여 콤퓨터가 현재 동작중이며 그 IP 주소에 있는가 하는것을 알수 있게 한다. ping 으로 망검사를 진행하는 기본방법은 다섯가지인데 이 지령을 차례로 수행해야 한다. 우선 지령창문을 열고 다음의 지령을 입력한다.

```
$ ping 127.0.0.1
```

이 주소는 이미 론의된 재귀순환주소이다. 망쏘프트웨어가 알맞게 설치되였다면 다음과 같은 련속적인 통보문흐름들이 화면에 표시된다.

```
PING 127.0.0.1 (127.0.0.1) from 127.0.0.1 : 56(84) bytes of data.
```

```
64 bytes from localhost.localdomain (127.0.0.1):  icmp_seq=0 tt1=255
   time=164 usec
64 bytes from localhost.localdomain (127.0.0.1):  icmp_seq=l tt1=255
   time=124 usec
64 bytes from localhost.localdomain (127.0.0.1):  icmp_seq=2 tt1=255
   time=167 usec
```

Ctrl+C 건반을 누를 때까지 이것들은 화면에 계속 표시된다. 만약 이 동작이 진행되지 않는다면 이 책의 범위를 벗어 나 망을 다시 설치해야 할것이다.

다음의 지령을 실행시켜 본다.

$ ping *ipaddress*

여기서 *ipaddress* 는 콤퓨터 IP 주소이다. 여기에 콤퓨터의 IP 주소를 대입한다. 다음의 화면표시는 IP 주소가 192.168.4.65 로 설정되였을 때이다.

```
PING 192.168.4.65 (192.168.4.65) from 192.168.4.65 : 56(84) bytes of
   data.
64 bytes from localhost. localdomain (192.168.4.65): icmp_seq=0 tt1=255
   time=169 usec
64 bytes from localhost.localdomain (192.168.4.65): icmp_seq=l tt1=255
   time=214 usec
64 bytes from localhost.localdomain (192.168.4.65): icmp_seq=2 tt1=255
   time=146 usec
```

건반 Ctrl+C 를 누른다는것을 상기한다. 이 지령이 실행되면 망기판에 정확한 IP 주소가 성과적으로 설정된다. 만약 이것이 실행되지 않는다면 netcfg 의 Interfaces 단락을 다시 검사한다.

다음은 아래의 지령을 입력한다.

$ ping *mycomputer*

여기서 *mycomputer* 는 콤퓨터의 호스트이름이다. 이 지령으로 실제적인 콤퓨터의 호스트이름을 대입한다.

```
PING mycomputer (192.168.4.65) from 192.168.4.65 : 56(84) bytes of
data.
64 bytes from mycomputer.localdomain (192.168.4.65): icmp_seq=0 ttl=255
time=163 usec
64 bytes from mycomputer.localdomain(192.168.4.65):icmp_seq=1 ttl=255
time=124 usec
64 bytes from mycomputer.localdomain (192.168.4.65):icmp_seq=2 ttl=255
time=216 usec
```

만약 이 지령이 실행되면 망기판의 IP 주소에 호스트이름이 성과적으로 결합된것으로 된다. 이 지령이 실행되지 않는다면 netcfg 나 /etc/hosts 파일의 Host 단락을 다시 검사한다.

국부적인 콤퓨터가 망우에서 작업할수 있다는것이 확인되면 다음단계는 관문콤퓨터에 대한 접속을 검사한다. 실례로 관문콤퓨터의 IP 주소가 192.168.4.24 라면 다음의 지

령을 실행한다.

$ ping 192.168.4.244

망에 대한 실제적인 관문 IP 주소를 대입한다. 만약 이것이 동작하면 앞에서 보여 준 화면과 류사한 화면이 현시된다. 그러나 이 지령으로는 실지 관문콤퓨터가 동작하고 있는지는 알수 없다.

이것을 검사하기 위하여 다음과 같은 ping 지령을 실행한다.

$ ping 63.86.158.42

여기서는 관문콤퓨터가 인터네트에 접속되였음을 전제로 한다. 이 지령은 앞에서 설명한 IP 주소를 가지는 Sybex Web 싸이트에 접속할수 있다. 이 지령이 동작하면 결과는 앞에서 보여 준것과 류사하다는것을 알수 있다. 또한 같은 지령으로 다른 인터네트의 Web 싸이트들에 접속할수도 있다.

마지막으로 DNS 가 적당히 구성되였는가를 보기 위하여 마지막 하나의 ping 지령을 실행한다.

$ ping www.sybex.com

여기서는 관문콤퓨터가 인터네트에 접속되였을 때를 전제로 한다. 만약 이 지령이 동작하면 www.sybex.com의 IP 주소와 이름을 리용한것을 제외하고는 앞에서 보여 준것과 비슷한 결과를 볼수 있다. 같은 형태의 지령으로 다른 인터네트의 Web 싸이트에 접속할수도 있다.

마지막단계는 이름탐색이 동작하고 있는가를 알아 보는것이다. 이것을 검사하기 위하여 nslookup 지령을 리용하여 기정 DNS 봉사기를 리용하고 있는 주콤퓨터이름이 무엇인가를 찾는다. 이를 검사하려면 www.yahoo.com을 찾아 볼수 있다.

$ nslookup www.yahoo.com

성과적으로 실행되였다면 nslookup 지령은 다음과 같은 결과들을 현시한다.

```
Server:    dul.paus.ch
Address:    194.209.60.97

Name:      www5.yahoo.com
Address:    204.71.177.70
Aliases:    www.yahoo.com
```

리용하고 있는 이름과 주소가 표시되는가를 확인하고(이것을 리용하여 DNS 봉사기들이 정확하게 구성되였는가를 확인할수 있다.) 그다음 이름탐색결과를 현시한다.

이 모든 단계들이 동작한다는것을 전제로 하여 완전히 기능적인 콤퓨터망의 접속을 알맞게 진행할수 있다. 만일 동작하지 않는다면 http://www.linuxdoc.org /HOWTO/Networking-HOWTO.html에 있는 Networking-HOWTO 문서를 보면 된다.

수동적인 망구성

이 절에서는 망을 구성할수 있는 일련의 대리단계들을 취급한다. 이 단계들은 이써네트망기판모듈이 설정되였다는것을 전제로 한다.

대면부설정

첫 단계는 리용하려는 콤퓨터망대면부를 능동화하는 단계이다. IP 주소 100.100.100.10 로서 망기판 eth0 을 능동화하려고 한다면 다음과 같이 ifconfig 지령을 리용하여 대면부를 능동화할수 있다.

```
$ ifconfig eth0 100.100.100.10 netmask 255.255.255.0 up
```

이 지령은 체계가 IP 주소 100.100.100.10 과 eth0 에 대한 망마스크 255.255.255.0 을 지정하도록 하며 우에서의 파라메터로서 대면부를 능동화한다.

대면부가 성과적으로 능동화되였는지 검사해 보려면 앞에서 고찰한 /sbin/ifconfig 지령을 리용하여 다음과 같이 가능한 대면부목록을 본다.

```
eth0      Link encap: Ethernet     HWaddr 00:C0:F0:0D:76:5A
          inet addr:100.100.100.10     Bcast:100.100.100.255
             Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST    MTU:1500    Metric:1
          RX packets:11239889 errors:0 dropped:0 overruns:0
          TX packets:16384520 errors:3 dropped:0 overruns:0
          Interrupt:9 Base address:0x300

Lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          UP LOOPBACK    RUNNING    MTU:3584    Metric:1
          RX packets:2142848 errors:0 dropped:0 overruns:0
          TX packets:2142848 errors:0 dropped:0 overruns:0
          col1isions:0 txqueue1en:0
```

이 목록은 일반적으로 적어도 재귀순환장치(장치 lo)의 사용가능한 일부 망장치들을 포함한다. 이 경우에 eth0 대면부를 찾는다. Red Hat 는 초기기동시에 재귀순환장치들을 구성하기 위한 일련의 정보들을 설정한다. 체계가 안전하게 동작하도록 하기 위하여 국부적인 재귀순환장치가 구성되였는가를 확인한다. 체계에 lo(재귀순환장치)가 설정되지 않았다면 다음의 ifconfig 지령으로 그것을 구성하고 능동화할수 있다.

```
$ ifconfig lo 127.0.0.1 up
```

이름봉사기(DNS)설정

다음의 단계는 DNS 봉사기로 알려 진 이름봉사기를 설정하는 단계이다. 그러므로 본문편집기를 리용하여 /etc/resolv.conf 파일을 편집해야 한다. 이 파일은 망에서 탈퇴하고 주콤퓨터이름에 대한 주소를 찾는데서 알아야 하는 모든것들을 체계에 알려 준다.

일반적으로 다음과 같이 resolv.conf 파일에는 적어도 두개의 행이 편집되여야 한다.

```
domain landegg.edu
nameserver 194.148.43.194
```

domain 행은 국부체계의 령역이름을 지정한다. 콤퓨터가 령역이 foo.bar 인 망우에 놓여있다면 이 행은 domain foo.bar 로 된다. 두번째 행은 리용하려는 주봉사기이름의 IP 주

소를 지적한다. 이것은 국부망우의 콤퓨터이거나 인터네트봉사제공기(ISP)에서 실행하는 봉사기, 인터네트상의 공동이름봉사기일수 있다. 망관리기는 이 정보를 리용자들에게 제공할수 있다.

많은 싸이트들은 여백을 메꾸기 위해 다중이름봉사기를 리용한다. 어떤 원인으로 콤퓨터가 정지되거나 사용할수 없게 되면 사용자는 대리봉사기를 리용하여 이름을 찾아보아야 한다. 다음과 같이 resolv.conf 파일에 여러개의 nameserver 행들을 포함시켜 여러개의 봉사기들을 설정할수 있다.

```
search landegg.edu
nameserver 194.148.43.194
nameserver 194.148.43.196
nameserver 194.148.8.10
nameserver 194.148.1.10
```

이와 같은 상태에서 처음에 렬거된 봉사기는 이름을 탐색할 때 처음으로 진행되는 것이다. 이 탐색이 실패하면 Linux 는 목록의 두번째 봉사기 등으로 이동한다. 렬거된 모든 이름봉사기들이 질문에 대한 응답에서 실패하면 Linux 는 탐색을 포기하며 이름탐색시도는 실패한다.

국부적인 주콤퓨터파일설정

리론적으로는 DNS 봉사기를 리용하여 국부망우의 주콤퓨터들의 이름을 포함하여 모든 이름을 탐색할수 있지만 이것은 아주 비효과적이다. 적당한 규모로 된 망의 콤퓨터이름을 탐색하기 위해 DNS 봉사기를 교대로 리용할 때 국부적인 주콤퓨터파일을 리용하여 이 탐색을 진행할수 있다.

국부적인 주콤퓨터파일은 체계에 위치하고 있고 주콤퓨터이름과 IP 주소를 전환시키는 목록을 포함한다. 국부적인 주콤퓨터파일을 리용하려면 먼저 적당한 본문편집기를 사용하여 /etc/host.conf 파일을 편집해야 한다. 이 파일은 Linux 에 이름을 어떻게 탐색하는가를 알려 주는데 국부적인 주콤퓨터파일을 능동화하기 위해 다음과 같은 두개의 행을 포함해야 한다.

```
order hosts,bind
multi on
```

첫번째 행은 이름을 탐색할 때 먼저 국부적인 주콤퓨터파일을 검사하여 /etc/resolv.conf 파일에 있는 지령들에 따라 DNS(bind 는 DNS 를 실행하는 Linux 데몬을 의미한다. 즉 Berkeley Internet Name Daemon 을 나타낸다.)를 검사한다.

다음은 /etc/hosts 라고 하는 국부적인 주콤퓨터들의 파일을 만들어야 한다. 그러자면 적당한 본문편집기를 리용하여 이 파일을 편집하고 렬거된 매 주콤퓨터에 대해 하나의 행입구점을 만든다. 그 입구점은 다음의 형식으로 되여 있다.

IP_Address hostname alias alias alias …

행의 매 단락(IP 주소와 주콤퓨터이름, 별명)은 하나 혹은 그이상의 공백으로 분리된다. 설명행은 하쉬표식(#)으로 시작되고 구조화에 리용할수 있고 더 큰 주콤퓨터파일들에 있는 입구점들을 설명한다.

실례를 들어 보자. 다음과 같은 4개의 입구점들이 주콤퓨터파일로부터 주어 졌다.

```
194.148.43.194      servl.1andegg.edu
194.148.43.195      apps.1andegg.edu       apps
194.148.43.196      serv3.1andegg.edu      serv3    www.1andegg.edu
194.148.43.215      officel5.1andegg.edu   officel5
```

모든 행들은 IP 주소로 시작되며 그다음 주콤퓨터이름과 주콤퓨터별명을 준다는것을 주의해야 한다. 별명을 요구하지 않는다. 더우기 많은 콤퓨터들에 대하여 별명은 완전한 령역이름이 아니라 짧은 형태의 주콤퓨터이름을 포함한다. 이것은 령역이름과 함께 완전한 주콤퓨터이름을 입력하지 않고 국부망우의 콤퓨터들을 호출할수 있게 만들어 진다. 이 실레에서 apps.landegg.edu 는 국부적으로 apps 로, officd15.landegg.edu 는 office15 로 호출할수 있다.

경로기설정

마지막단계는 콤퓨터망과 그밖의 세계와 대화하도록 하여 주는 필수적인 경로기를 설정하는것이다. 기본적으로는 매 대면부에 대해서 Linux 에 어떤 콤퓨터망이 접속되였는가를 알려 주는 경로입구점이 필요하다. 더우기 콤퓨터망이 관문을 통하여 다른 비국부적인 망들과 접속되였다면 기정관문을 나타내는 입구점이 요구된다.

재귀순환장치를 가지고 시작해 보자. 재귀순환장치를 통해서 국부적인 콤퓨터가 호출된다는것을 지정하기 위해서는 route 지령을 리용해야 한다. 즉

```
$ /sbin/route add -host 127.0.0.1 lo
```

이 지령은 주콤퓨터 127.0.0.1(국부망의 주콤퓨터)을 장치 lo 를 통해서 호출하여야 한다는것을 보여 준다.

다음은 앞에서 언급한 eth0 을 고찰해 보자. 이 장치는 개인용콤퓨터를 망주소 100.100.100.0 와 망마스크 255.255.255.0 을 가진 망에 접속시킨다. 다시 route 지령을 리용하여 eth0 대면부에서 국부망에 대한 모든 정보를 내보내도록 Linux 에 알려 준다.

```
$ /sbin/route add -net 100.100.100.0 netmask 255.255.255.0 eth0
```

마지막으로 다시 한번 route 지령을 사용하여 원격망을 호출하는데 리용할수 있는 기정관문을 지정한다. 만약 기정관문이 IP 주소 100.100.100.1 에 있다면 다음과 같은 지령을 리용한다.

```
$ /sbin/route add default gw 100.100.100.1 eth0
```

이것은 기정관문이 주소 100.100.100.1 에 있으며 eth0 대면부를 통해서 관문에 대하여 예정된 정보를 보내는 방법으로 호출할수 있다는것을 표시한다.

마지막으로 파라메터와 인수가 없는 route 지령을 리용하여 모든 경로선택입구점들을 검사한다. 이것은 다음과 같은 경로조종표를 복귀시킨다.

```
Kernel IP routing table
Destination    Gateway        Genmask        Flags  Metric  Ref  Use   Iface
100.100.100.0  *              255.255.255.0  U      0       0    317   eth0
127.0.0.1      *              255.0.0.0      U      0       0    6     1o
default        100.100.100.1  0              UG     0       0    2605  eth0
```

초기기동시 자동적인 콤퓨터망구성

지금까지 완전히 기능적인 콤퓨터망접속방법들을 보았다. 그러나 Linux체계가 기동할 때마다 매번 건반으로 모든 지령들을 입력시키는것은 힘들다.

다행히도 /etc/rc.d/rc.local 와 같은 체계의 시동스크립트들중의 하나에 필요한 ifconfig 과 route 지령 모두를 배치할수 있다. 여기서 리용되는 실례에서는 본문편집기를 리용하여 rc.local 파일에 다음과 같은 행을 추가하면 된다.

```
/sbin/ifconfig eth0 100.100.100.10 netmask 255.255.255.0 up
/sbin/ifconfig 1o 127.0.0.1 up
/sbin/route add -host 127.0.0.1 1o
/sbin/route add -net 100.100.100.0 netmask 255.255.255.0 eth0
/sbin/route add default gw 100.100.100.1 eth0
```

Linux/Unix 망에서의 파일공유

Linux 체계는 국부적인 TCP/IP 망에서 다른 콤퓨터들과 대화할수 있으므로 하나의 가장 기초적인 콤퓨터망과제 즉 파일공유를 보기로 한다. Unix 와 Linux세계에서 일반적으로 이것은 NFS(Network File System)를 리용하여 실현한다.

공유는 두가지 방향에서 진행한다.

다른 콤퓨터들로부터 콤퓨터망우의 다른 주콤퓨터로부터 공유된 등록부들과 파일들을 호출

다른 콤퓨터에로 국부망우의 개인용콤퓨터들로부터 파일들과 등록부들을 공유

콤퓨터망에서 등록부공유

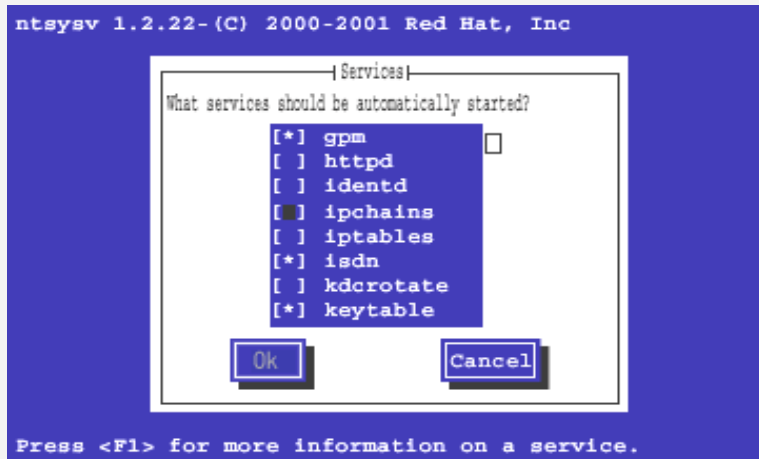
망에서 등록부들을 호출하기전에 그것을 공유해야 한다. NFS를 리용하여 등록부들을 반출하려면 두가지 데몬 즉 /usr/sbin/rpc.nfsd, /usr/sbin/rpc.mount와 함께 하나의 구성파일 /etc/exports이 요구된다. rpc.mountd 와 rpc.nfsd파일은 대부분의 체계에 설치된다. 만약 이 파일이 체계에 없다면 Red Hat Linux 7.1 CD-ROM에 있는 RPM 파일 nfs-utils-0.3.1-5.i386.rpm을 체계에 설치하여야 한다.

망파일체계와 방화벽(NFS와 Firewalls)

Red Hat Linux 7.1을 설치할 때 기정인 방화벽을 설치하였다면 그것은 망파일체계를 통해서 등록부들을 공유하는것을 막는다. 만일 콤퓨터가 국부망안에 있고 다른 망들과 접속한것이 없다면 국부적인 콤퓨터에서 방화벽을 불가능으로 할수 있다. 방화벽기능은 오직 콤퓨터가 다른 망에 접속되였을 때만 필요하다(aka, 관문콤퓨터).

/etc/rc.d/init.d/ipchains stop나 /etc/rc.d/init.d/iptables stop지령중 어느 하나를 써서 방화벽을 불가능으로 할수 있다. 그러나 체계를 재시동할 때까지는 효력이 없다. Ntsysv차림표를 통해서도 앞으로 방화벽을 불가능으로 할수 있다. /usr/sbin/ntsysv지령을 실행한다. Services차림표가 나타날 때 다음의 화면에서 ipchains와 iptable의 봉사를 선택하지 않는다.

공유하려고 하는 등록부들이 망과 인터네트에 접속된 관문콤퓨터상에 존재한다

면 방화벽이 필요하다. 지어 방화벽이 있다면 NFS 를 보호로 리용하지 말아야 한다. 그러나 여전히 NFS 를 리용하여 관문콤퓨터로부터 등록부들을 공유하려고 한다면 제 31 장 《보안과 값 눅은 경로기로서의 Red Hat Linux 7.1》에 서술된 방화벽구성편의프로그람들을 리용하여 포구 111 과 2049 를 열어야 한다. 그러면 망파일체계가 요구하는 통신통로들을 열수 있다.

그다음 본문편집기를 리용하여 /etc/exports 파일을 편집하는 방법으로 콤퓨터망의 등록부를 공유할수 있다. 공유된 매 등록부에 하나의 입구점을 추가한다. 입구점들은 다음의 형식으로 되여 있다.

/directory host (options)

다른 말로 말하면 등록부를 공유할 때 지정된 사용자들이 아니라 지정된 콤퓨터들과 그것을 공유한다. /etc/exports 의 안내서페지에서 /etc/exports 파일의 실례를 아래에 보여 준다.

```
# sample /etc/exports file
/                master(rw) trusty(rw,no_root_squash)
/projects        proj*. local .domain(rw)
/usr             *.local .domain(ro)
/home/joe        pc001(rw,all_squash, anonuid=150,anongid=100)
/pub             (ro,insecure,a11_squash)
/pub/private     (noaccess)
```

이 파일을 한행씩 고찰해 보자.

1 행 이 행은 하쉬마크(#)로 시작되는 설명행이며 무시된다.

2 행 두개의 콤퓨터(master 와 trusty)에는 /등록부에 대한 호출준위가 각이하게 주어 진다. 즉 master는 읽기와 쓰기권한을 가지며 trusty는 읽기와 쓰기권한과 함께 수출된 모든 파일과 보조등록부들에 완전한 호출을 할수 있는 뿌리사용자권한을 가진다(no_root_squash 항목). 다른 말로 trusty의 뿌리사용자는 국부체계에서 뿌리와 같은 권한을 가진다.

3 행 이름이 proj 로 시작되고 부분령역 local.domain 인 주콤퓨터는 등록부 /projects 에 대하여 읽기와 쓰기권한을 준다.

4 행 local.domain 령역안에 있는 주콤퓨터는 /usr 등록부에 대해서 읽기전용(ro 항목)호출속성을 가진다.

5 행 주콤퓨터 pc001 은 등록부 /home/joe 에 대해서 읽기쓰기권한을 가진다. pc001 콤퓨터로부터 이 등록부를 호출하는 모든 리용자들은 닉명사용자(all_squash 선택)로서 취급되며 닉명사용자들은 150 인 리용자 ID 와 150 인 그룹 ID 를 가진것으로 고찰된다(anonuid=150 과 anongid=150 항목).

6행 등록부 /pub는 반출하는 체계에 대한 호출로서 모든 주콤퓨터들을 사용가능으로 만든다. 호출은 모든 닉명사용자들과 함께 읽기전용으로 허가된다. insecure 항목은 NFS 용으로 예약된 TCP/IP 포구를 리용하지 않는 의뢰기들이 그 설치를 호출하도록 한다.

7 행 등록부 /pub/private 에 대해 모든 주콤퓨터들에로의 호출은 거부된다(noaccess 항목).

/etc/exports 파일에 대한 다른 항목들은 exports 의 안내서페지에서 찾아 볼수 있는데 지령은 다음과 같이 리용한다.

```
$ man exports
```

일단 /etc/exports 에서 수출되는 등록부를 지정하면 rpc.nfsd 와 rpc.mountd 처리를 다시 해야 한다(보통 초기기동시에 실행한다.). 우선 ps 를 리용하여 두개의 처리공정의 처리공정 ID 를 검사한다.

```
$ ps aux  |  grep rpc.nfsd
root 1103    0.2  0.8 872  508  ?  S  22:25  0:00  /usr/sbin/rpc.nfsd
$ ps aux  |  grep rpc.mountd
root 1105    0    0.7 836  500  ?  S  22:25  0:00 /usr/sbin/rpc.mountd
```

다음은 처리공정을 금지시키고 다시 시작한다.

```
$ kill 1103 1105; /usr/sbin/rpc.nfsd; /usr/sbin/rpc.mountd
```

경고 우에서 보여 준 ps aux 지령을 자체로 실행한다. 우에서 보여 준 ps aux 지령을 실행하여 화면에 출구한 수자들은 대체로 1103 과 1105 로서 서로 다르다. 적당히 대입한다.

새로운 수출들은 망에서 원격사용자들이 호출할수 있게 한다.

일러두기 여러개의 Linux 배포물들에서 **rpc.nfsd** 와 **rpc.mountd** 를 재시동하기 위해서는 바로 **nfs** 데몬을 재시동해야 한다. 실례로 Red Hat Linux 7.1 에서 뿌리사용자로서 **/etc/rc.d/init.d/nfs restart** 지령을 실행한다.

원격파일체계호출

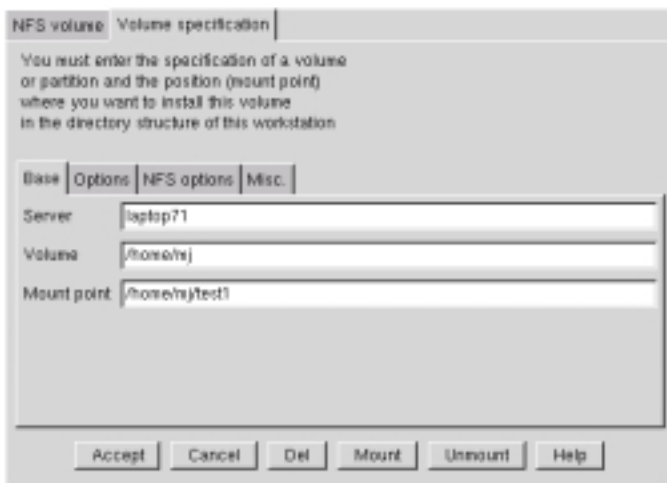
원격파일과 등록부들에 대한 호출은 mount 지령을 리용하여 진행한다. 그 지령형식은 다음과 같다.

$ mount *remote-directory-name local-directory*

여기서 원격등록부는 *hostname:/directory-name* 의 형식으로 지정되며 국부등록부는 현재 존재하는 등록부이거나 빈 등록부로서 이것들은 원격등록부를 호출하는 경로로 된다.

실례를 들어 보자. 국부망에서 콤퓨터 foo.bar의 등록부 /test/dir를 호출하며 등록부는 NFS를 거쳐서 원격설치가 가능하게 되였다고 가정한다. 콤퓨터에 foo라고 부르는 하나의 빈 등록부가 있는데 이것은 원격등록부에 대한 설치지점이다.

이 등록부에 접속하기 위해 다음의 지령을 리용한다.

```
$ mount foo.bar:/test/dir /foo
```

이 지령을 실행하면 국부등록부 /foo에 foo.bar콤퓨터의 /test/dir 등록부를 접속시킬수 있다. 일단 이 지령을 수행하면 파일에 할당된 허가권한을 리용해서 원격등록부의 파일들과 보조등록부를 호출할수 있다. 실제적으로 원격등록부 /foo 의 내용들은 foo.bar 콤퓨터의 등록부 /test/dir 내용들이다.

경고 망파일체계 NFS에서 주의할것이 한가지 있다. 국부적인 체계에서 뿌리사용자는 파일에 대해 누가 만들고 소유했는가에 관계없이 체계상의 아무데서나 열고 읽고 삭제하는 무제한한 권한을 가진다. 그러나 원격으로 설치된 등록부안에서 뿌리는 NFS 모형하에서 여러가지 제한을 가지게 된다. 즉 명백하게 권한을 부여 받은 파일들과 등록부들만을 호출한다.

초기기동시 원격등록부들을 설치

자주 리용하는 원격등록부들이 있다면 아마 그것들을 초기기동시에 자동적으로 설치하려고 할것이다. 이를 위해서 Linuxconf 지령을 리용한다. linuxconf-auth 지령과 함께 Linuxconf 를 시작한다. 구성화면의 목록에서 Config 표쪽안에서 File Systems ⇨ Access NFS Volume을 선택한다. 이 화면에는 현재 NFS설치목록들이 들어 있다. Add 단추를 찰칵하여 그림 28-13에서 보여 주는 NFS Volume Specification 화면을 현시한다.

그림 28-13. NFS Volume Specification 화면

우에서 취급한 원격설치실례들을 리용하기 위하여 Server 마당에는 **laptop71**, Volume 마당에는 **/home/mj** 그리고 Mount Point 마당에는 **/home/mj/test1** 를 건입력한다. 초기기동시 자동적으로 그것이 설치되게 하려면 Options 표쪽을 선택해서 Not Mount At Boot Time 이 선택되지 않았는가를 확인한다. 마지막에 자동설치구성을 끝내기 위하여 Accept 단추를 찰칵한다.

그러면 파일 /etc/fstab 에 다음과 같은 입구점들을 추가한다.

linux71:/home/mj   /home/mj/test nfs   exec,dev,ro 1 1

만약 체계기동시 설치해야 할 자동설치를 수동적으로 생성하려면 본문편집기를 리용해서 체계에 있는 /etc/fstab 파일에 류사한 입구점을 추가할수 있다.

기초적인 망보안

콤퓨터망의 접속과 그 모든 자원의 호출범위가 크다면 이것은 현재의 망보안에 아주 위태롭다. 그렇지만 다행히도 Linux 에는 망에 가입된 콤퓨터들이 Linux 체계상에서 접속되여 봉사 받으려 하는것을 조종하는 기초적인 기구를 제공한다. 기초적인 보안기구들은 2개의 파일로 이루어 졌다. 즉 /etc/hosts.allow 파일과 /etc/hosts.deny 파일이다.

이것들은 다음과 같은 3개 부분으로 된 기본적인 보안사슬을 이룬다.

1. hosts.allow 파일안에 있는 데몬-의뢰기결합에 접근하게 한다.
2. hosts.deny 파일안에 있는 데몬-의뢰기결합에 대한 호출을 거부한다.
3. 기타 모든 사용자에게 호출권한을 준다.

기정으로 파일 hosts.allow 혹은 hosts.deny 안에는 입구점들이 없기때문에 이 경우는 항목3이다. 즉 누구나 체계에 련결할수 있다. Telnet 나 FTP 와 같은 독특한 봉사기를 설정하여 보다 강력한 인증을 요구할수 있다고 하더라도 알려 지지 않은 체계를 통한 자유분망한 해커들의 침입은 막을수 없다.

hosts.allow 와 hosts.deny 들의 입구점작성은 간단하다.

Daemon_List: Client_List

이와 같은 입구점들은 망데몬(실례로 ftpd, in.telnetd, in.rshd 등등)이 어느 주콤퓨터에 접속해야 하며(hosts.allow) 혹은 어느 주콤퓨터에 접속하지 말아야 하는가를 지정한다(hosts.deny). 어떤 망데몬들은 목록화할수 있다. 실례로 hosts.allow 파일안의 입구점

ftp: host1

은 지정된 주콤퓨터인 host1 로부터 FTP 에 들어 오는 접속을 허가한다.

몇개의 특정한 예약어들은 모든 데몬들과 개별적인 의뢰기들을 목록화함으로써 작업을 보다 쉽게 해준다. ALL 예약어는 의뢰기나 데몬을 정합할수 있다. 실례로 다음의 hosts.allow 에 있는 입구점

ALL: ALL

은 임의의 의뢰기로부터 호출을 임의의 형태로 하도록 하며 반대로 hosts.deny 파일안에서 같은 입구점들은 모든 호출을 금지한다.

EXCEPT 예약어는 한편 ALL 에 의해서 광범히 공개된 망에서 례외하도록 한다. 실례로 hosts.deny 파일안에서 입구점

ALL EXCEPT ftpd: ALL

은 FTP 봉사를 제외하고 임의의 봉사를 호출하는 모든 의뢰기들에 대한 접근을 금지한다. 류사하게 Hosts.allow 파일안에서 입구점

ALL: .local.domain EXCEPT foo.local.domain

은 foo.local.domain 을 제외하고 령역 local.domain(반드시 local.domain 앞에 있는 "."을 주의한다.)에 있는 임의의 콤퓨터에 대한 모든 봉사를 호출하도록 한다.

마지막으로 LOCAL 예약어는 국부령역의 모든 콤퓨터들에 대한 호출봉사를 쉽게 한다. 즉 hosts.allow 파일에서

ALL: LOCAL

은 국부령역안의 콤퓨터들로부터 임의의 형태로 호출하도록 한다.

hosts.allow 와 hosts.deny 파일들은 대부분 사용자들의 요구보다 더 좋은 조종들을 제공하는 아주 풍부한 입구점들의 묶음을 취할수 있다. 이것은 hosts.allow 파일의 사용설명서페지에서 상세히 취급한다.

```
$ man hosts.allow
```

▶미리 알아두기

이 장에서는 TCP/IP 를 리용하여 Ethernet 망과 Linux PC 의 접속에 대한 기초를 학습하였다. 이것은 다른 Linux 체계를 포함하면서 Linux 가 망우의 다른 Unix 체계와 쉽게 통신하도록 한다.

그러나 Linux 체계가 Windows 와 Novell 망과 완전히 통합되자면 보다 더 많은 문제들을 풀어야 한다. 제 29 장 《Windows 와 Novell 망에서 Red Hat Linux 7.1 의 통합》에서 Windows 와 Novell 과 Linux 체계를 통합시키는 기초에 대하여 보게 된다.

제 29 장. Red Hat Linux 7.1 을 Windows 와 Novell 망에 통합

이 장에서는 Linux 가 현실에 존재하는 많은 망들에 추가할수 있는 능력을 시험하게 된다.

표준 Windows 망을 실행하고 있는 경우에 Linux 는 효률적이고 능률적이며 강력한 파일봉사기 및 인쇄봉사기의 역할을 담당할수 있다. Windows NT/2000 보다는 원가가 훨씬 적고 Windows 98/95/Me 보다는 훨씬 안전한것으로 하여 Linux 는 망우에서 Windows 탁상들사이에 파일공유를 필요로 하는 운영비지출사무에서의 리상적인 파일봉사기 및 인쇄봉사기로 될수 있다.

다른 방향에서 고찰해 보면 Linux 는 자기의 체계상에 있는 국부적인 하드디스크처럼 Winodws 파일봉사기상의 파일들을 호출할수 있다. 이것은 몇가지 목적으로 봉사한다. 실례로 Linux 기초의 Web 봉사기를 실행하는 경우에도 Web 개발자는 Windows 체계상에서 개발을 진행한다. Linux Web 봉사기는 Web 봉사기의 내용을 마음대로 수정하지 못하게 하면서 Windows 개발체계에 미리 정해 진 등록부와 자동적으로 접속되며 Web 봉사기에 새로운 파일을 복사할수 있다.

다음으로 이 장에서는 Novell 망에 Linux 체계를 통합하는 방법을 고찰한다. 여기서 리용되는 도구들은 Linux 를 Windows 망에 통합할 때 리용한 도구들만큼 원만히 개발되지 못하고 견고하지도 못하지만 이 장에서는 Novell 선택항목을 론의하고 론점에 따르는 정보원천을 준다.

Windows 망과의 Linux 파일공유와 인쇄기공유

Linux-Windows 망통합의 가장 일반적은 방법은 Linux 체계를 망의 Windows 의뢰기에 대한 파일봉사기 및 인쇄봉사기로서 동작하게 한것이다. 이것은 Samba 를 리용해서 실현할수 있다. Samba 는 Linux 가 Windows 망의 파일 및 인쇄기를 공유하는데서 기초로 되는 SMB(봉사기통보문블로크)규약을 쓸수 있게 해주는 쏘프트웨어묶음이다. Linux 에 기초한 SMB 봉사기의 가능한 의뢰기에는 LAM Manager, Windows 95/98/Me, Windows NT/2000, OS/2, 기타 Linux 체계들이 속한다.

Samba 의 설치

이 책과 함께 제공된 Red Hat Linux 7.1 CD 에는 완전한 Samba 프로그람제품이 없다. 그러나 두장의 Red hat Linux 7.1 설치 CD-ROM 으로 Linux 를 설치하면 Samba 도 같이 설치된다. rpm 지령으로 이것을 확인할수 있다.

```
$ rpm -q samba
$ rpm -q samba-client
$ rpm -q samba-common
```

Samba 가 설치되지 않았으면 론리적인 위치(/mnt/cdrom 과 같이)에 CD-ROM 을 설치하여 적합한 Red Hat CD-ROM 으로부터 그것을 설치한 다음 rpm 을 리용하여 세개의 Samba 프로그람묶음을 설치할수 있다. 즉

```
# rpm -i /mnt/cdrom/RedHat/RPMS/samba-2.0.7-36-i386.rpm
# rpm -i /mnt/cdrom/RedHat/RPMS/samba-client-2.0.7-36.i386.rpm
# rpm -i /mnt/cdrom/RedHat/RPMS/samba-common-2.0.7-36.i386.rpm
```

주의 이 책과 함께 제공된 CD-ROM 에는 samba-client 와 samba-common 프로그람묶음만이 있다. 그러므로 국부망의 Microsoft Windows 콤퓨터사이에서는 공유자원을 충분히 리용할수 있지만(다음절에서 보게 되는 Smbmout 지령을 참고해야 한다.) 같은 망에 있는 Linux 콤퓨터와 다른 Microsoft Windows 쏘프트웨어사이에서는 자원을 충분히 공유할수 없다.

인터네트에 의한 Samba 의 설치

Samba 의 최신판본을 얻으려면 최신판본의 Samba 원천코드나 RPM 을 내리적재하여야 한다. 그 반사싸이트를 찾으려면 Samba Web 싸이트 http://www.samba.org를 참고하여야 한다. 미국에 있는 반사싸이트중의 하나는 http://us1.samba.org/samba/ftp/이다.

요구되는 파일들을 내리적재한 다음 압축된 파일을 풀고 원천코드로부터 samba 를 어떻게 콤파일하고 설치하겠는가를 설명한 문서를 조사하여야 한다.이러한 처리과정은

판본마다 차이가 있기때문에 여기서는 그것들에 대한 내용을 서술하지 않았다.

한편 원천코드로부터 콤파일하는것이 자신이 없거나 두장으로 된 Red Hat 7.1 CD가 없는 경우에 가장 간단하게 설치할수 있는 방법은 ftp.redhat.com 혹은 www.rpmfind.net와 같은 원천지로부터 rpm 파일들을 내리적재하는것이다. 그다음 알맞는 rpm –i 지령을 리용하여 그것들을 설치한다.

설치하면 무슨 파일이 얻어 지는가?

Samba 를 설치하면 다음과 같은 파일이 얻어 진다.

/usr/sbin/smbd 의뢰기로부터의 접속을 조종하는 Samba 프로그람

/usr/sbin/nmbd Windows 망의 봉사기에 접속할수 있는 의뢰기를 위한 NetBIOS 봉사기

/usr/bin/smbclient SMB 봉사기를 호출하기 위한 기본 Samba 의뢰기

/etc/sambasmb.conf Samba 구성파일

Samba 의 구성

전체의 Samba 구성은 현재 /etc/samba/smb.conf 파일에 있다. Smb.conf 파일은 여러개의 파라메터들에 따르는 머리부를 포함하는 여러개의 항목들로 구성된다. 실례로 다음의 항목은 그 망에 있는 /tmp 등록부를 공유한다.

```
[tmp]
    comment = Temporary file space
    path = /tmp
    read only = no
    public = yes
```

머리부는 중괄호안에 있으며 매개 파라메터는 선으로 구별되고 파라메터의 형식은 파라메터=값으로 표시된다.

[global]부분

대부분의 smb.conf 파일들은 Samba 의 전체 동작에 영향을 주는 본질적인 파라메터들을 정의하는 [global]부분으로 시작한다. [global]부분에서 공통파라메터들로는 다음과 같은것들이 있다.

printing 리용하려는 인쇄체계의 형을 지정한다. 낡은 Linux 배포물이 아니면 이 형은 lprng 로 된다.

printcap name printcap 파일의 위치를 지정한다. 일반적으로 /etc/printcap 으로 된다.

load printers 인쇄기를 공유한 경우에 yes 로 표시된다.

quest account 어느 Linux 사용자가 손님으로 될수 있는가를 지정한다. 대부분의 Linux 관리기들은 극히 제한된 특혜를 주기 위해 이것을 nobody 로 한다.

workgroup Samba 봉사기가 속하는 작업그룹이나 령역을 지정한다.

security 체계자원에 대한 사용자호출권한을 지정한다. 여기에는 user, server, domain, share 의 4 가지가 있다.

user Samba 는 자기 콤퓨터의 /etc/passwd 파일로부터 사용자이름과 통과암호를

확인한다. 이것은 가장 일반적이고 가장 쉬운 선택항목이다.

server Samba는 password server 파라메터에 의하여 지정된 Winodws NT/2000 령역봉사기로부터 확인한다. 다른 Samba봉사기는 Windows NT/2000 도메인봉사기를 모의하기 위하여 설정될수 있다.

domain 적합한 Windows령역조종기에 Samba봉사기의 NetBIOS이름을 추가할것을 요구한다. 대표적으로 15문자이하인 경우 NetBIOS이름은 주콤퓨터이름과 같다.

Share Microsoft Windows-style workgroup의 고유한 형태로서 Linux를 동위망으로 설정하기 위하여 리용하는 파라메터이다. NetBIOS 이름은 주콤퓨터이름과 같아야 하고 15문자보다 작아야 한다.

대표적으로 쓰이는 smb.conf 파일의 [global]부분은 다음과 같다.

```
[global]
    printing = lprng
    printcap name = /etc/printcap
    load printers = yes
    quest account = nobody
    workgroup = testgroup
    security = user
```

이 부분은 인쇄체계의 형을 지정하고 Linux 사용자로서 nobody 를 사용하는 손님등록자리(quest account)를 설정하며 Windows 작업그룹으로서 testgroup 의 봉사기부분을 구성하며 Samba 가 인증을 위하여 자기콤퓨터상의 /etc/passwd 파일을 사용한다는것을 지정한다.

[homes]부분

[homes]부분은 사용자의 홈등록부를 Windows 망에로 어떻게 공유하는가를 정의하는데 리용된다. 후에 보게 되겠지만 이 부분이 없으면 사용자의 홈등록부들을 개별적으로 공유해야 한다.

smb.conf 파일에서 다음의 [homes]부분을 보자.

```
[homes]
     comment = Home Directories
     browseable = no
     read only = no
     preserve case = yes
     short preserve case = yes
     create mode = 0750
```

comment 파라메터는 그 부분에 대한 추가적인 정보를 주지만 조작에는 영향을 주지 않는다. browseable 파라메터는 다른 사용자들이 등록부를 열람할수 있는가 지적하는데 리용된다. 즉 허가 받은 그 사용자들이 호출을 제한하기 위해서 no 로 설정한다. read only 파라메터는 사용자들이 자기의 홈등록부에서 파일을 읽기도 하고 쓰기도 할수 있도록 하기 위해서 no를 설정한다. preserve case 파라메터와 short preserve case 파라메터는 Linux가 예민하기때문에 민감하지 못한 Windows체계가 파일이름의 문자들을 변경시키지 못한다는것을 담보한다. Creat mode 파라메터는 Samba를 통해 접속된 사용자들이 만든 파일에 대한 허가를 규정한다. 이 실례에서 0750 은 소유자가 파일들을 읽고

쓰고 실행할수 있으며 소유자와 같은 그룹에 있는 사용자들이 알고 실행할수 있지만 기타 사용자들에게는 금지된다는것을 의미한다.

사용자는 Windows 체계로부터 자기의 홈등록부를 호출하기 위하여 공유이름을 위한 자기의 사용자이름을 리용해야 한다. 실례로 username 의 사용자가 smbserv 에 있는 자기의 홈등록부를 호출하기 위하여 \\ smbserv\ username 공유를 호출하게 된다.

[printers]부분

[homes]부분이 모든 사용자들의 홈등록부를 공유하는데 리용된것처럼 [printers]부분도 Windows 망에로 모든 인쇄기들을 공유할수 있게 한다.

표 29-1 에 [printers]부분에 있는 파라메터에 대한 설명을 주었다.

표 29-1. [printers]부분의 일반적인 파라메터

파라메터	설 명
Comment	이 부분에 대한 정보를 표시하며 조작에는 영향을 미치지 않는다.
Path	완충통로를 지정한다. 즉 사용자소유의 Samba 를 위한 완충등록부를 만들고 리용할수 있다(실례로 Red Hat Linux 에서는 /var/spool/samba 를 만들고 리용한다).
Browseable	홈등록부에서처럼 이 파라메터를 no 로 설정해 놓으면 허가된 사용자들만이 인쇄기를 리용할수 있다.
Printable	Yes 로 설정하지 않으면 인쇄작업을 할수 없다.
Public	만일 yes 로 설정되면 여러개의 망에서 손님은 인쇄할수 있는 권한을 가지며 손님사용자가 인쇄실행을 방해하지 못하도록 한다.
Writable	인쇄기를 쓸수 없게 한다. 따라서 이 파라메터는 no 로 설정한다.
Create mode	이 파라메터는 인쇄중에 만들어 진 완충파일에 대한 허가를 규정한다. 주로 0700 으로 설정한다.

종합해 보면 이 파라메터들은 다음의 실례에서 보여 준 표준 Red Hat Linux 에서 아주 일반적인 [printers]부분을 만들수 있다.

```
[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = no
    printable = yes
    public = no
    writable = no
    create mode=0700
```

Windows 로부터 Linux 인쇄기들을 호출하면 등록부와 같은 방법으로 작업한다. 공유이름은 printcap 파일에 있는 Linux 인쇄기이름이다. 실례로 smbserv 우에서 printname 을 가진 인쇄기를 호출하기 위하여 사용자들은 \\ smbserv\ printname 을 호출한다.

공개호출등록부의 공유

때때로 모든 사용자들이 읽기전용으로 호출할수 있는 공개적인 등록부를 만드는것이 필요하다. 이 등록부는 정보의 변경으로부터 사용자들을 보호하는 방법으로 정보를

공유하도록 설정할수 있다.

다음의 항목을 고찰해 보자.

```
[public]
    path = /public/directory
    public = yes
    read only = yes
    printable = no
```

이 항목은 모든 사용자들이 호출가능한 공유이름 public 를 생성하는데(public = yes) 이것은 읽기전용(read only = yes)이다.

다른 실례를 고찰해 보자. 모든 사용자들이 읽고 쓸수 있는 공개적인 림시등록부로서 /tmp 를 공유한다. 이것은 다음의 항목에서 볼수 있다.

```
[tmp]
    path = /tmp
    read only = no
    public = yes
```

비공개호출등록부의 공유

공개호출등록부를 공유하는것외에도 제한된 사용자들이 호출가능한 공유등록부를 만들수도 있다.

공유이름 private 로 /private/directory 등록부를 공유하려는 위치를 고찰해 보자. 이 등록부가 3명의 사용자 즉 user1, user2, user3만이 호출할수 있는것이라고 하자. 이것은 다음의 항목과 류사하다고 볼수 있다.

```
[private]
    path = /private/directory
    valid users = user1 user2 user3
    public = no
    writable = yes
    printable = no
    create mask = 0765
```

여기에 새로운 파라메터 valid users 가 나왔다. 이 파라메터는 그 값으로서 자원에로 호출할수 있는 사용자들의 목록을 취한다. 사용자이름은 공백으로 분리된다.

public 는 마음대로 들여 다 보지 못하도록 no 로 설정하며 writable 은 완전히 쓰기가능한 자원으로 만들기 위하여 yes 로 설정한다.

모든 파라메터의 종합

이제 모든 파라메터들을 smb.conf 파일에 구성해 보자.이 파일을 편집할 때 주의해야 할것은 다음과 같다.

- 공백행들은 무시된다.
- 설명행은 반두점 혹은 #기호로 시작된다.

종합된 파일은 다음과 같다.

```
; Sample smb.conf
; Global Settings
[global]
    printing = bsd
    printcap name = /etc/printcap
    load printers = yes
;The guest user is nobody
    quest account = nobody
    workgroup = testgroup
    security = user
; Export All Home Directories to the Network
[homes]
    comment = Home Directories
    browseable = no
    read only = no
    preserve case = yes
    short preserve case = yes
    create mode = 0750
; Make All Printers Accessible on the Network
[printers]
    comment = All Printers
    path =  /var/spool/samba
    browseable = no
    printable = yes
    public = no
    writable = no
    create mode = 0700
; Create Public read-only Directory
[public]
    path = /public/directory
    public = yes
    read only = yes
    printable = no
; Provide a Public Temporary Directory
[temp]
    path = /tmp
    read only = no
    public = yes
; Export a Private Workspace for user1, user2 and user3
```

```
[private]
    path = /private/directory
    valid users = user1 user2 user3
    public = no
    writable = yes
    printable = no
    create mask = 0765
```

Samba 실행

콤퓨터상에 있는 망이 TCP/IP로 실행되고 있다면 이 구성을 완성하여 Samba를 실행할 준비를 갖춘다. 그러나 Net BIOS 망이 실행되고 있으면 여러가지 준비작업이 필요하다. /etc/services 파일에서 다음의 행들을 찾아 보아야 한다(만일 그것들이 없으면 추가해야 한다.).

```
netbios-ns      137/tcp     # NETBIOS Name Service
netbios-ns      137/udp
netbios-dgm     138/tcp     # NETBIOS Datagram Service
netbios-dgm     138/udp
netbios-ssn     139/tcp     # NETBIOS session service
netbios-ssn     139/udp
```

이제는 Samba 가 다 구성되였으므로 그 쏘프트웨어를 실행할수 있다. 기정으로 samba의 Red Hat Linux 판본은 항상 리용상태에 있어야 하므로 Linux 기동시에 기동될수 있도록 설치해야 한다. 또한 체계스크립트 /etc/rc.d/init.d/smb는 samba를 수동으로 기동시키거나 중지시킬수 있다. 실례로

```
# /etc/rc.d/init.d/smb start
```

지령으로 Samba를 기동할수 있다.

그리고 다음의 지령으로 중지할수 있다.

```
# /etc/rc.d/init.d/smb stop
```

또는 다음의 지령으로 smbd와 nmbd를 동시에 기동하고 중지시킬수 있다.

```
# /etc/rc.d/init.c/smb restart
```

다음 두개의 지령은 뿌리사용자가 수동으로 Samba를 기동하는데 리용할수 있다.

```
/usr/sbin/smbd -D
/usr/sbin/nmbd -D
```

Linux 체계로부터 Windows 망파일과 인쇄기에로의 접근

이 처리과정의 간단한 측면은 Linux가 파일과 인쇄기를 공유하는 SMB를 호출할수 있게 한것이다. 이를 실현하는 방법에는 여러가지가 있다. 가장 간단한 방법은 Samba를 설치해 주는 두개의 의뢰기프로그람인 smbclient와 smbprint을 리용하는것이다.

이 방법은 얼마간 제한이 있는데 특히 그것은 파일호출에서 나타난다. Smbclient는 원격공유를 호출하는 FTP와 같은 방법을 제공한다. 물론 이것은 파일조작을 위한 지령

들인 cp 나 mv 와 같은 표준 Linux/Unix 지령의 사용을 허가하지 않기때문에 다른 응용프로그람들로부터 공유파일을 호출하지 못한다(Linux 응용프로그람이 국부파일체계처럼 보이는 NFS 형의 원격파일체계와는 다르다.).

이 문제는 사실상 NFS 파일체계, 국부파일체계와 같이 Linux 에 설치된 파일체계들을 공유할수 있게 하는 SMB 와 같은 체계를 허용하는 smbfs 프로그람묶음을 리용함으로써 해결할수 있다.

Smbclient 의 리용

smbclient 프로그람은 보통 /usr/bin 에 설치된다. 이 프로그람은 FTP 와 같은 대면을 리용하며 SMB 봉사기에 있는 공유자원으로부터 혹은 그 공유자원에로 파일들을 옮기는데 리용한다.

Smbclient 사용의 첫 단계는 SMB 봉사기상의 자원에 접속하는것이다. 가장 단순한 지령형식은 다음과 같다.

$ smbclient //*server*//*resourname*

물론 간단하지 않은 경우도 있다. 실례로 보호된 자원에 대해서 통과암호를 리용하는 경우에는 지령이 좀더 복잡해 진다.

$ smbclient //*server*/*resourname password*

이외에도 smbclient 가 봉사기에 접속하는 방법을 바꾸는 여러가지 기발들이 있다. 그 주요기발들을 표 29-2 에 보여 주었다.

표 29-2. 일반 smbclient 기발

기 발	설 명
-L 호스트	이 기발은 주콤퓨터로 지정된 봉사기에서 리용할수 있는 자원들의 목록을 표시한다. 즉 이 기발을 리용하면 자원을 지정할 필요가 없다.
-I IP 주소	이 기발은 IP 주소는 알고 있지만 봉사기의 주콤퓨터이름을 모르는 경우 쓸모 있다. 즉 smbclient 는 그 콤퓨터가 지정된 IP 주소를 가진것으로 본다.
-N	이 기발은 통과암호재촉문을 금지시킨다. 자원이 통과암호를 요구하지 않을 때 리용한다. 통과암호가 요구되지 않을 때 이 기발을 지정하지 않으면 사용자는 빈 통과암호를 주기 위하여 Enter 건을 누른다.
-U 사용자이름	이 기발을 리용하여 자원에 접속하기 위한 사용자이름을 지정할수 있다. 이 기발이 지정되지 않으면 봉사기는 USER 혹은 **LOGNAME** 환경변수의 내용을 리용한다. 만일 이것도 비여있으면 사용자이름은 봉사기에 제공되지 못하게 된다. 사용자이름뒤에 퍼센트기호(%)를 붙임으로써 봉사기에 따르는 통과암호를 보낼수 있으며 다음과 같이 표시할수 있다. **-U 사용자이름%통과암호**
-w 작업그룹	이 기발은 봉사기에 접속할 때 어느 작업그룹이 리용되는가를 지정한다.
-T tar 선택항목	이 기발은 국부 Linux 체계에서 자료를 **tar** 파일에 넣거나 꺼낼 때 리용한다. 실례로 **-Tx backup.tar 는 backup.tar** 로부터 원격공유에로 파일들을 꺼낼 때 리용한다. 그리고 **-Tc backup.tar** 는 원격공유안에 있는 모든 파일들과 등록부들을 포함하는 **tar** 파일인 **backup.tar 를** 만든다.

실례로 이 정보에 기초한 smbclient 지령을 만들어 보자.

$ smbclient //server/resourcename -U username%password -W workgroup

이 지령은 username 과 password 를 가진 사용자가 workgroup 에 속해 있는 봉사기상에서 resourcename 에 접속하기 위한것이다.

파일자원에 대한 리용가능한 조작

일단 콤퓨터가 파일자원에 접속하면 자료를 여기저기로 옮길수 있는 많은 지령들이 있다. 이 지령들을 표 29-3 에 보여 주었다.

표 29-3. 파일조작지령

조 작	설 명
cd 등록부	SMB 공유자원상에서 다른 등록부로 옮겨 간다.
del 파일	봉사기상에서 지정된 파일을 지운다(rm 지령을 리용할수 있다.).
dir	봉사기로부터 현재등록부의 목록을 표시한다(ls 지령을 리용할수 있다.).
get 파일	원격봉사기로부터 지정된 파일을 얻어서 국부체계에 있는 현행등록부에 같은 이름으로 보관한다. 선택적으로 국부체계상에서 파일이름을 다르게 줄수도 있다. 즉 **get 파일. 국부파일**
lcd 등록부	국부체계의 현행등록부를 변경시킨다.
mget 파일마스크	원격봉사기상에서 지정된 파일마스크와 일치하는 모든 파일들을 얻는다.
mkdir 등록부	원격봉사기우에 지정된 등록부를 만든다(md 지령을 쓸수도 있다.).
mput 파일마스크	지정된 파일마스크와 일치하는 국부등록부상의 모든 파일들을 원격봉사기의 현행등록부에 넣는다.
prompt	다중파일조작(mput 와 mget)에서 프롬프트를 절환하는 기능이다. On 으로 설정하면 사용자들은 매개 파일이 복사될 때 마다 지령을 대기한다.
put 파일	현재의 국부등록부에서 지정된 파일을 현재의 원격봉사기등록부에로 같은 파일이름으로 복사한다. 원격파일이름을 바꾸려면 **Put 파일 원격파일이름**지령을 리용할수 있다.
quit	**smbclient** 를 끝낸다(**exit** 지령을 쓸수도 있다.).
recurse	다중파일조작(mput 와 mget)에서 등록부순환을 절환하는 기능이다. On 으로 설정되면 파일들을 복사할 때 현재등록부아래의 모든 등록부들을 거쳐 검색한다.
rmdir 등록부	원격봉사기에서 지정된 등록부를 지운다(**rd** 지령을 리용할수도 있다.).

몇가지 실례를 들어 보자.

- 국부등록부를 보조등록부 foo 로 바꾸려면 지령 lcd foo 을 리용해야 한다.
- 원격등록부를 ./foo 로 바꾸려면 지령 cd../foo 을 리용해야 한다.

- 국부등록부안에 있는 파일 foo 를 원격공유에로 복사하면서 파일이름을 newfoo 로 바꾸는 지령은 put foo newfoo 이다.
- 원격공유의 현재등록부에서 .txt 로 끝나는 모든 파일들을 얻으려면 지령 mget *.txt 을 리용해야 한다.
- 원격공유상에 새로운 등록부 foo 를 만들려면 지령 mkdir foo 을 리용해야 한다.

인쇄기자원에 대한 리용가능한 조작

smbclient 를 리용하여 인쇄기자원에 접속하면 다음의 지령들을 인쇄기작업에 리용할수 있다.

Print 파일 Printmode 로 지정된 방식에 기초하는 현재자원을 리용하여 지정된 파일을 인쇄한다.

Printmode 선택항목 지정된 선택항목에 기초한 인쇄기의 방식을 설정한다. 가능한 선택항목으로서는 graphics 혹은 text 가 있다. 여기서 graphics 방식은 2 진자료형식을 지정한다.

Queue 원격인쇄대기렬의 현재상태를 표시한다.

quit(혹은 exit) smbclient 를 끝낸다.

물론 이것은 좀 다루기 어려운 조작이다. 본문파일을 하나 인쇄하자고 해도 보통 Unix 인쇄대기렬을 리용하면 간단히 조작할수 있는 작업을 여기서는 쉽게 할수 없다.

대신에 smbclinet 로 인쇄기에 접속해서 <print mode 본문>지령과 <print 파일이름> 지령을 실행할수 있다. 좀더 복잡한 인쇄형식으로 인쇄하는것은 더 어렵다.

실례로 만일 원격인쇄기가 PCL 인쇄기와 그에 리용되는 쏘프트웨어를 가진것이라면 어떻게 PostScript 만을 출구하겠는가(Unix 세계에서는 종종 그런 경우가 있다.)? 이런 경우에 사용자는 어떤 파일을 인쇄하는것이 필요하며 PCL 방식으로 그것을 변환하기 위하여 gs 를 통하여 이 파일을 통과시키며 smbclient 로 인쇄기에 접속하고 인쇄방식을 설정하고 파일을 인쇄한다. 이것은 적지 않은 작업량이다.

다행히도 smbprint 가 이 문제를 해결할수 있다.

smbprint 의 리용

smbprint 스크립트는 표준 Linux/Unix 인쇄기대기렬에 대하여 smbclinet 지령을 적용하여 인쇄할수 있게 하는 지령이다. 즉 smbprint 는 원격 SMB 인쇄기자원에로 가상적으로 사용자들을 련결하지 않고 인쇄할수 있게 한다.

대부분의 Samba 설치프로그람이 있으면 /usr/bin/smbprint 에 있는 스크립트를 볼수 있다.

이 스크립트를 리용하기 위하여 원격인쇄기를 위한 printcap 항목을 만들고 그 인쇄기가 어디에 있는가를 지정하는 구성파일을 작성한다.

먼저 printcap 항목을 보기로 하자.

```
queuename:\
        :sd=/var/spool/samba:\
        :af=/var/spool/samba/accountingfile:\
        :if=/usr/bin/smbprint:\
        :mx=0:\
        :lp=/dev/null:
```

여기에서 중요한 항목만을 고찰해 보자.

Sd=/var/spool/samba 완충등록부를 지정한다.
Af=/var/spool/samba/accountingfile 등록자리파일을 지정한다. 이 파일은 구성파일과 같은 등록부 즉 자기의 완충등록부에 있다.
If=/usr/bin/smbprint 입력려파프로그람이 smbprint 이라는것을 지정한다.
/p=/dev/null 인쇄기가 printcap 파일에 있는 콤퓨터와 물리적으로 련결되지 않는다는것을 나타낸다.

다음은 등록자리파일과 같은 등록부에 있는 .config 라고 하는 구성파일을 작성해야 한다. 이 파일에는 세개의 항목이 있는데 이것은 다음과 같다.

server=*SERVERNAME*

service=*PRINTERNAME*

password="*password*"

콤퓨터의 이름을 인쇄기가 *SERVERNAME* 을 단 콤퓨터이름으로 바꾸고 인쇄기의 이름을 인쇄기가 *PRINTERNAME* 을 단 콤퓨터이름으로 바꾼다. 일단 이것이 끝났으면 거의 모든 파일들은 lpr 지령을 리용하여 인쇄할수 있어야 한다.

$ **lpr -p** ***queuename filename***

인쇄를 조종하는 Netscape 6 과 같은 응용프로그람이므로 이 구성은 많은 인쇄기들과 응용프로그람으로 작업하게 된다.

smbfs 의 리용

smbfs 프로그람묶음은 Linux 에서 NFS 를 설치하는것과 마찬가지로 SMB 공유 Linux 에로 직접 설치하게 한다. 일단 smbfs 가 설치되면 그 리용은 아주 간단하다. Red Hat Linux 7.1 에서는 samba-2.0.7-36.i386.rpm 프로그람묶음에 SMBFS 가 포함되여 있다.

Smbmount 의 리용

smbfs 체계에서 기본은 smbmount 프로그람이다. smbmount 프로그람은 Linux 체계에 SMB 공유를 설치하는데 리용되는 도구이다. smbmount 의 일반형식은 다음과 같다.

smbmount ***//servername/resourcename mountpoint***

mount 지령과 마찬가지로 설치위치는 그 체계에 존재하는 등록부(빈 등록부)이여야 한다. 일부 쉘들에서 거꿀빗선을 쓰는 문제를 피하기 위하여 SMB 공유에서는 거꿀빗선 대신에 빗선을 리용한다.

경고 **smbmount** 는 봉사기이름을 조사하는데 NetBIOS 를 쓸수 없다. 그러므로 SMB 봉사기이름이 그 봉사기의 TCP/IP 주콤퓨터이름과 차이나면 쓸수 없다. 이 경우에 문제의 그 봉사기를 위한 Linux/Unix 주콤퓨터이름을 리용한다.

smbmount 지령에 대한 완전한 문서는 smbmount 사용설명서페지이다.

Linux를 Novell망에 접속

견고성의 견지에서 Netware-Linux 결합망은 Windows-Linux 결합망보다 훨씬 못하다. 이런것으로해서 이 절에서는 리행과정에 대해서 상세하게 보지 않고 리용가능한 선택항목만을 고찰한다.

주의	무료로 리용할수 있는 쏘프트웨어령역에서 **ncpfs**라고 하는 Novell Netware의뢰기프로그람제품의 리용범위는 상당히 제한되여 있다. 그러므로 이것은 여전히 **mars_nwe**라고 하는 제 2의 Netware봉사기제품으로 된다. 두 제품은 다 자기들이 지원하는 판본과 서로 호환할수 있는 Netware 제품형태에 대해서 제한된다. 두 제품은 Red Hat Linux 7.1에 포함되여 있다. 콤퓨터에 IPX를 설치하는것이 필요할수도 있다. IPX 편의프로그람묶음은 이 책과 함께 제공된 Red Hat Linux 7.1 CD-ROM 에서 리용할수 있는데 이것은 **ipxutils-2.2.0.18-3.i386 rpm으로서 /RedHat/RPMS**보조등록부에 있다.

ncpf와 mars_new는 다같이 완전한 무료Linux Netware문제해결을 위한 가장 밀집한 도구를 제공한다.

ncpfs프로그람제품은 ftp.gwdg.de/pub/linux/misc/ncpfs에서, mars_new프로그람묶음은 http://www.compu-art.de/download/mars-nwe에서 내리적재할수 있다. 특히 mars_new 에 대한 문서의 대부분이 도이췰란드에 있는데 같은 지령은 **IPX-OWTO**에서 쓸수 있다.

그러나 두 제품들은 제한성을 가진다. 실례로

- 이 제품들은 둘다 NetWare 3.x 호환방식에서만 쓸수 있으며 모방방식에서는 NetWare 4.x와 5.x만을 리용할수 있다.
- ncpfs는 NetWare 2.x봉사기와 거꿀호환성을 가지지 않는다.
- ncpfs는 Windows NT 3.51과 같이 일부 NetWare호환봉사기에서는 작업하지 못한다.

▶미리 알아두기

이 장에서 Linux와 Windows사이에 파일을 공유할수 있는 Windows인트라네트에 Linux를 통합하는 방법을 배웠다.

제 30장 《Red Hat Linux 7.1과 DOS/Windows》에서는 오늘날 공동세계에서 중요한 문제인 DOS와 Windows 쏘프트웨어를 Linux에서 실행하는 기능을 고찰하게 된다.

세계를 놀래우는 Linux는 dosemu 프로그람을 리용하여 DOS 응용프로그람들을 지원하고 있다. 16bit와 32bit Windows 응용프로그람들을 실행할수 있는 능력을 가진 Wine 이라고 부르는 Windows 모의프로그람을 쓰는것은 아직도 시기상조이지만 노력하면 방법이 있다. 같은 환경에서 매일매일 작업할수 있는 도구를 설치할 준비가 되여 있지 않다 하더라도 Wine 프로그람은 현실적으로 Microsoft Word나 다른 일반적인 Windows응용프로그람들을 실행할수 있다. 제 30장에서는 두개의 량립되는 프로그람묶음들의 특성과 리용방법을 보게 될것이다.

또한 이장에서는 가상기계를 설정하게 하는 VMware프로그람과 상업적목적을 가진

응용프로그람들을 시험한다. 여기서는 Microsoft Windows 와 Linux 둘중에서 선택되는 조작체계를 실행할수 있다.

제 30 장. Red Hat Linux 7.1 과 DOS/Windows

문서처리와 같은 생산성과제들에서 일상적인 조작체계로 Linux를 리용하는것을 반대하는 가장 큰 론거의 하나가 Linux에서 리용할수 있는 응용프로그람들이 적다는것이다. 그러나 Linux에서 Windows응용프로그람들을 실행할수 없다는 론거는 없으며 있다면 기이한 사실로 된다.

이 론거에서 주목되는 문제는 Linux가 혼자서 DOS의 주요응용프로그람들과 많은 Windows응용프로그람들을 실행할수 있다는 사실이며 따라서 Windows와 넓은 범위에서 호환성을 기대할수 있다. 이러한 기대를 현실화하는것이 바로 VMware와 Win4Lin라는 두개의 업무용응용프로그람들이다. 이 프로그람에 의해서 Linux에서 Microsoft계 Windows응용프로그람들을 실행할수 있다. Win4Lin프로그람은 Microsoft Windows 95/98의 응용프로그람만 실행할수 있지만 VMware프로그람은 Microsoft계 Windows의 모든 판본들을 Linux에서 실행시킬수 있는 기능을 가지고 있다. VMware프로그람은 또한 Microsoft계 Windows NT/2000에서 Linux의 모든 판본들을 실행시킬수도 있다. 이 장에서는 Linux를 위한 VMware워크스테이션들을 설치하는 기초적인 방법에 대하여 보게 된다.

DOS가 지원하는 응용프로그람들은 한때 Linux에서 리용되여 왔으며 많은 대중적인 DOS응용프로그람들은 크게 품을 들이지 않고도 Linux에서 실행시킬수 있다.

또한 Linux환경에서 Windows와 완전한 호환성을 제공하는 현실적인 수단들도 있다. 기본적인 수단으로서 Wine을 들수 있다. Linux개발팀들은 업무적인 Windows 프로그람묶음보다도 좋은 Wine을 무료로 리용할수 있도록 공개하는 사업에 착수하였다. 이 사업에 리해관계를 가지는 업무용프로그람개발자들의 도움으로 Wine은 현재 Linux에서 Microsoft계 Windows응용프로그람들을 고속으로 실행할수 있게 하는 도구들을 가진 프로그람제품으로 발전하고 있다.

Linux 에서 DOS 응용프로그람의 실행

DOS응용프로그람에 대한 지원기능은 Linux에서 제공하는 강력한 호환성이며 이 호환성은 거의 모든 Linux배포물들에 들어 있다.

DOSEmu환경에서 Linux는 DOS 6.2와 같은 판본이나 Windows 95/98 DOS를 비롯한 DOS판본들을 설치하고 실행할수 있으며 그때 많은 DOS응용프로그람들을 실행할수 있다. 물론 일부 도형과 서체조작에 영향을 주는것을 비롯하여 제한성이 좀 있지만 전체적으로 보면 프로그람들이 잘 실행된다.

이밖에도 DOS환경은 X Windows의 각이한 창문에서 실행할수 있으며 Linux프로그람을 실행하면 DOS응용프로그람들을 실행하기가 쉽다.

DOSEmu가 제공하는 특성들은 다음과 같다.

- DOS응용프로그람들을 동시에 실행하는 다중사용자기능
- DOS환경에서 규정된 구동기문자로 Unix등록부들을 설치하는 기능
- Unix인쇄기로 인쇄하는 기능

DOSEmu 설치

DOSEmu는 표준 Red Hat Linux 7.1배포물에는 없기때문에 다른 곳에서 가져 와서 설치해야 한다. 판본이 1.0.1인 최신판본배포물은 http://www.dosemu.org/stable에 있는 DOSEmu홈페지의 압축된 문서고(tar.gz)와 RPM양식들에서 리용할수 있다.

만일 Red Hat Linux 7.1(혹은 RPM프로그람제품을 리용하는 다른 배포물)을 리용한다면 DOSEmu를 가져다가 설치할수 있다. 실례로 그것을 /tmp등록부에 끌어다 놓고 뿌리사용자방식으로 가입해서 다음과 같은 지령으로 설치한다.

```
rpm –i /tmp/dosemu-1.0.1-1.i386.rpm
```

DOSEmu 구성

Linux에서 DOSEmu를 실행시키기 위하여서는 많은 구성정보들을 설정해야 한다. 그것들은 다음과 같다.

- 플로피디스크에 의한 DOS의 기동
- 디스크영상에 의한 DOS의 기동
- 분리된 파티션에 의한 DOS의 기동

이 장에서는 프로그람이 어떻게 동작하는가를 보기 위하여 디스크영상으로 DOS를 기동하는것만 본다. 다른 매체로 기동하는 방법들을 보여 주는 보다 개선된 구성정보들은 문서에 충분히 기록되여 있다. DOSEmu를 설치한후에 /usr/doc/dosemu/등록부에서 이 문서를 찾아 볼수 있다.

주의 DOSEmu에 대한 구체화된 정보들과 문서는 http://www/dosemu.org에서 볼수 있다. 이것은 http://www.dosemu.org/docs/HOWTO/의 HOWTO에 있다. 이 싸이트로부터 완전한 설치지령을 리용하여 DOSEmu의 최신판본을 내리적재할수 있다.

DOS를 실행하기 위해서는 /etc/dosemu.conf라고 하는 구성파일이 있어야 한다. 이것은 확장문서를 가진 매우 복잡한 파일이다.

DOSEmu프로그람묶음은 디스크영상으로부터 기동하도록 설계된 디스크영상파일과 dosemu.conf파일이 구성되여 있으면 쉽게 작성할수 있다. 디스크영상파일 그자체는 /var/lib/dosemu/hdimage에 위치하고 있다. 이 파일에는 fdisk, format, unix2dos , lredir(Linux등록부를 DOS구동기문자로 전환하는 편의프로그람)를 비롯한 많은 지령과 도구들이 있다. 파일에는 또한 FreeDOS로 알려 진 DOS의 변종들도 있다.

전형적인 DOSEmu환경을 구성할 때 어떤 정보들을 설정하는가를 보기 위하여 기정 dosemu.conf파일의 내용을 볼수 있다.

```
###########################################################################
#This file is /etc/dosemu.conf, included by /var/1ib/dosemu/global .conf
#
# Linux DOSEMU configuration for parser versions >= 3 (dosemu-0.97.1)
#
# ./doc/README.txt (chapter 2.) contains a description of the syntax
# and the usage of dosemu.conf.
#
#
```

```
# Access rights are defined in
#
#       /etc/dosemu.users
#
############################################################################


# Notes for editing this section:
#
#   In    $_xxx = (n)     n is a numerical or boolean value
#                  =       =
#   In    $_zzz = "s"     s is a string
#
# Please edit only between the brackets and quotes and keep the rest
intact.
#             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^^^^^^^^^^^^^^^^^^^^

$_debug = "-a"                  # same format as -D commandline option
                                # (but without the -D in front)
$_features= ""                  # list of temporary hacks, see release notes in
                                # the file ChangeLog. e.g "0:1 2:0", which
                                # means
                                # to set feature_0 to 1 and feature_2 to 0.
$_timint = (on)                 # emulate INT08 type timer interrupts
$_mathco = (on)                 # or off
$_cpu = (80386)                 # CPU emulation, valid values:  80[345]86
$_rdtsc = (off)                 # if possible use Pentium cycle counter
$_cpuspeed = (0)                # 0 = calibrated by dosemu, else given
                                  (e.g.166.666)
$_pci = (off)
$_xms = (1024)                  # in Kbyte
$_ems = (1024)                  # in Kbyte
$_ems_frame = (0xe000)
$_dpmi = (off)                  # in Kbyte
$_dosmem = (640)                # in Kbyte, < 640
$_hardware_ram = ""             # list of segment values/ranges such as
                                # "0xc8000 range 0xcc000,0xcffff"
$_secure ="ngd"                 # secure for: n (normal users), g (guest),d
                                  (dexe)
```

```
                                # empty string: depending on 'restricted'
                                # "0": always insecure (not recommended)
$_odd_hosts = ""                # black list such as "lucifer.hell.com
                                  bi1ly.the.cat"
$_disk1ess_hosts=""             # black list such as "hackerl newbee
# gateway1"
$_emusys = ""                   # empty or 3 char., config.sys   ->
# config.XXX
$_emubat = ""                   # empty or 3 char., autoexec.bat ->
                                # autoexec.XXX
$_emuini = ""                   # empty or 3 char., system.ini   ->
# system.XXX
$_hogthresho1d = (1)            # 0 == all CPU power to DOSEMU
$_irqpassing = ""                # list of IRQ number (2-15) to pass to
                                # DOS such
                                  as
#"3 8 10"
$_speaker = ""                  # or "native" or "emulated"
$_term_char_set = ""            # Global code page and character set
                                # selection
                                # "" == automatic, else: ibm, latin, latin1,
                                  latin2
$_term_co1or = (on)             # terminal with color support
$_term_updfreq = (4)            # time between refreshs (units: 20 == 1
                                # second)
$_escchar = (30)                # 30 == Ctrl-,^ special-sequence prefix
$_rawkeyboard = (0)             # bypass normal keyboard input, maybe
                                # dangerous
$_1ayout = "auto"               #one of: finnish(-latinl),de(-latinl),be,it,
                                  us
                                # uk, dk(-latinl), keyb-no,no-latin1,dvorak,
                                  po
                                # sg(-latinl), fr(-latinl), sf(-latinl),
                                  es(-latinl)
```

```
                              # sw, hu(-latin2), hu-cwi, keyb-user
                              # hr-cp852, hr-latin2, hu-cwi, keyb-user
                              # or 'auto' (which tries to generate the
                              # table
                              # from the current Linux console settings)
$_keybint = (on)              # emulate PCish keyboard interrupt
$_X_updfreq =(5)              # time between refreshs (units: 20 == 1
                              # second)
$_X_title = "DOS in a BOX"    # Title in the top bar of the window
$_X_icon_name = "xdos"        # Text for icon, when minimized
$_X_keycode = (auto)          # on == translate keybord via dosemu
                              # keytables
                              # or 'off or ' auto'
$_X_blinkrate = (8)           # blink rate for the cursor
   $_X_font = ""              # basename from/usr/X11R6/lib/Xll/
                              # fonts/misc/*
                              # (without extension) e.g. "vga"
$_X_mitshm = (on)             # Use shared memory extensions
$_X_sharecmap = (off)         # share the colormap with other applications
$_X_fixed_aspect = (on)       # Set fixed aspect for resize the graphics
                                window
$_X_aspect_43 = (on)          #Always use an aspect ratio of 4:3 for graphics
$_X_lin_filt = (off)          # Use linear filtering for >15 bpp interpolation
$_X_bilin_filt = (off)        # Use bi-linear filtering for >15 bpp
                                interpolation
$_X_mode13fact = (2)          # initial size factor for video mode 0x13
                                (320x200)
$_X_winsize = ""              # "x,y" of initial windows size (defaults to
                                float)
$_X_gamma = (1.0)             # gamma correction
$_X_vgaemu_memsize = (1024)     # size (in Kbytes) of the frame buffer for
                                  emulated vga
$_X_lfb = (on)                # use linear frame buffer in VESA modes
$_X_pm_interface = (on)       # use protected mode interface for VESA modes
```

```
$_X_mgrab_key = ""          # KeySym name to activate mouse grab,
                            # empty == off
$_X_vesamode = ""           # "xres,yres ... xres.yres"
                            # List of vesamodes to add.The list has to
                              contain
                            # SPACE separated "xres,yres" pairs
$_video = "vga"             # one of: plainvga, vga, ega, mda,mga, cga
$_conso1e = (0)             # use 'console' video
$_graphics = (0)            # use the cards BIOS to set graphics
$_videoportaccess = (1)     # allow videoportaccess when 'graphics'
                                # enabled
$_vbios_seg = (0xc000)      # set the address of your VBIOS (e.g.
                                # 0xe000)
$_vbios_size = (0x10000)    # set the size of your BIOS (e.g. 0x8000)
$_vmemsize = (1024)         # size of regen buffer
$_chipset = ""              # one of: plainvga, trident, et4000,diamond,
                                  avance
                                # cirrus, matrix, wdvga, paradise, ati,
                                # s3, sis
$_dua1mon = (0)             # if you have one vga _p1us_ one hgc (2
                                  monitors)
$_vbootfloppy = ""          # if you want to boot from a virtual floppy:
                                # file name of the floppy image under
                                  /var/1i b/dosemu
                                # e.g. "floppyimage" disables $_hdimage
                                #       "floppyimage +hd" does _not_
                                #       disable
                                           $_hdimage
$_f1oppy_a ="threeinch"     # or "fiveinch" or "atapi" or empty, if not
                                  existing
                                # optionally the device may be appended
                                # such as
                                # "threeinch:/dev/fdo"
$_f1oppy_b = ""             # dito for B:
```

```
$_hdimage = "hdimage.first"  # list of hdimages under /var/lib/dosemu
                               # assigned in this order such as
                               # "hdimage_c hdimage_d hdimage_e"
                               # If the name begins with '/dev/', then
                               # partition
                               # access is done instead of virtual hdimage
                               #such
                               ➡as
                               # "/dev/hdal" or "/dev/hdal:ro" for
                               # readonly
                               # Currently mounted devices and swap are
                               # refused.
                               # Hdimages and devices may be mixed such
                               # as
                               # "hdimage_c /dev/hdal /dev/hda3:ro"
                               # Note: 'wholedisk' is _not_ supported.
$_hdimage_r= $_hdimage         # hdimages for 'restricted access(if different)
$_aspi = ""                    # list of generic SCSI devices to make
                               # available
                               # for the builtin aspi driver (format of an
                               # entry
                               # is 'device:type:mappedtarget' such as
                               # "sg2:WORM sg3:Sequential-Access:6 sg4
                               # CD-OM"
                               ➡or
                               # "sg2:4 sg3:1:6 sg4:5" (which are equal)
$_coml = ""                    # e.g. "/dev/mouse" or  "/dev/cua0"
$_com2 = ""                    # e.g. "/dev/modem" or "/dev/cual"
$_com3 = ""                    # dito                  "/dev/cua2"
$_com4 = ""                    # dito                  "/dev/cua3"
$_ttylocks = ""                # Lock directory (e.g. "/var/lock")
                               # default ("") is /usr/spool/uucp
$_mouse = ""                   # one of: microsoft,mousesystems,logitech,
                               ➡ mmseries
                               # mouseman, hitachi, busmouse, ps2
```

```
$_mouse_dev = ""                # one of: com1, com2, com3, com4 or
                                # /dev/mouse
$_mouse_f1ags = ""              # list of none or one or more of:
                                # "emu1ate3buttons cleardtr"
$_mouse_baud = (0)              # baudrate, 0 == don't set
$_printer = "1p"                # list of (/etc/printcap) printer names to
                                ➡appear as
                                # LPT1, LPT2, LPT3 (not all are needed,
                                #empty
                                ➡for none)
$_printer_timeout = (20)        # "idle time in seconds before spooling out
$_ports = ""                    # list of portnumbers such as "0xlce 0xlcf
                                ➡0x238"
                                # or "0x1ce range 0x280,0x29f 310"
                                # or "range 0xla0,(0xla0+15)"
$_ipxsupport = (off)            # or on
$_novell_hack = (off)
$_vnet = (off)                  # 'on' for packet-multi (used by
                                # dosnet)
$_sound = (off)                 # sound support on/off
$_sb_base = (0x220)
$_sb_irq = (5)
$_sb_dma = (1)
$_sb_dsp = "/dev/dsp"
$_sb_mixer = "/dev/mixer"
$_mpu_base = "0x330"
```

다행히도 재구성은 많이 요구되지 않는다. 현재DOSEmu(1.0.1)판본으로 하드디스크영상과 기동등록부를 설정하고 그다음 일부 표준MS-DOS기동파일을 새로운 기동등록부에 복사해야 한다.

/var/lib/dosemu/등록부에 있는 다음과 같은 지령으로 하드디스크영상(hdimage.first)과 기동등록부(/var/lib/dosemu/bootdir.first)를 설정할수 있다. 즉

$ **/var/lib/dosemu/setup-hdimage**
$ **/var/lib/dosemu/setup-bootdir**

아쉽게도 FreeDOS프로그람은 DOS모방기로부터 Linux파일들과 등록부들에 대한 관리를 지원하지 못한다. 이것을 해결하려면 3개의 기초적인 DOS구성파일들이 필요하다. 실례로 표준MS-DOS 혹은 PC-DOS기동디스크를 가지고 있다면 COMMAND.COM, IO.SYS 그리고 MSDOS.SYS파일을 새로 만든 /var/lib/dosemu/bootdir.first 등록부에

복사한다.

이 세개의 파일들은 /dev/fd0 장치(A구동기)에 설치되지 않은 플로피디스크에 있다고 본다. Linux는 예민하기때문에 이 3개의 파일들이 FreeDOS판본에 있는것과 겹쳐 기록되도록 다음의 지령을 리용해야 한다.

```
$ mcopy a:io.sys /var/lib/dosemu/bootdir.first/io.sys
$ mcopy a:command.com /var/lib/dosemu/bootdir.first/command.com
```

주의 표준MS-DOS기동플로피디스크 판본 4.0.1과 그이상인 체계파일에서 앞에서 이야기한 파일들을 복사할수 있다. 이 체계파일들은 Windows 95/98기동디스크처럼 DOS의 일부 다른 판본들에서 실행할수 있다.

끝으로 홈등록부안에 .dosemurc구성파일을 작성하거나 편집한다. 이 파일에는 적합한 DOS구성파일과 Linux를 대응시키는 하나의 행이 있다. 즉

```
$_hdimage = "bootdir.first"
```

DOS기동

DOSEmu를 실행시키는 가장 간단한 방법은 console 혹은 xterm창문으로부터 dos 지령을 리용하는것이다. 즉

```
$ dos
```

DOSEmu는 /etc/dosemu.conf파일에서 정의된 기동장치로부터 적재되고 실행된다. 기정으로 이것은 제공된 디스크영상파일로부터 기동된다는것을 의미한다. 그림 30-1은 dos지령을 실행한후에 xterm 창문에 나타난 DOS지령재촉문을 보여 준다.

그림 30-1. xterm창문에서 DOS를 실행

통보문이 어떻든 상관없이 이 결과는 DOSEmu1.0.1에 기초한다. 일단 DOS가 실행되면 표준DOS지령과 플로피디스크로부터 프로그람을 실행시키기 위한 문법 혹은 이 장의 마지막에서 보게 되겠지만 전환된 등록부에 포함된 응용프로그람들을 실행시키기 위한 문법 등을 리용할수 있다.

개별적인 X 창문에서 DOS의 기동

현재창문에서 DOS를 실행시키는 방법외에 X Winodws를 실행할 때 X 창문에 DOS 상태를 나타나게 할수 있다. 다음과 같은 지령을 입력하는 방법으로 그것을 수행한다.

```
$ xdos
```

실질적으로 xdos지령은 기본dos 2진파일과의 련결이다. xdos련결을 통하여 프로그람을 실행하는것은 다음의 지령으로도 할수있다.

```
$ dos -x
```

자체의 X창문에서 DOS를 실행시키면 그림 30-2와 같은 결과가 나타난다.

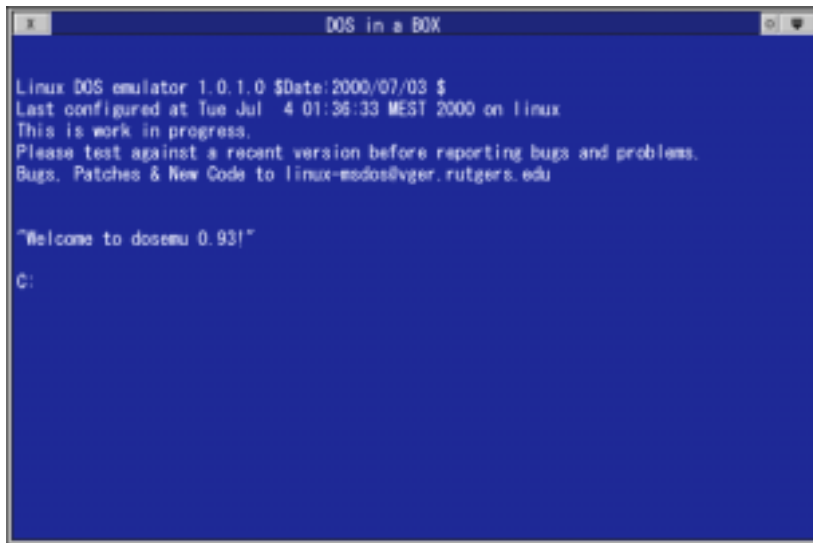



그림 30-2. 자체의 x창문에서 DOS실행

DOSEmu지령의 리용

DOSEmu를 포함하고 있는 디스크영상에는 표 30-1에서 보여 준것들을 비롯하여 여러개의 지령들과 편의프로그람들이 있다. 이 지령들과 기타 많은 편의도구들은 기정디스크영상을 리용해서 DOSEmu를 실행시킬 때 C:\ bin과 C:\ dosemu등록부에서 찾아 볼수 있다.

표 30-1. 선택된 DOSEmu지령

지 령	설 명
eject.com	CD-ROM구동기에서 CD-ROM을 뽑는다.
emumouse.com	DOSEmu마우스구동프로그람을 조종한다.
exitemu.com	DOSEmu를 끌낸다.
lredir.com	Unix등록부를 DOS구동기문자로 절환한다.
unix.com	DOSEmu로부터 Linux지령을 실행한다

lredir 지령에 의한 등록부의 설치

DOSEmu를 리용할 때 그것은 디스크영상 혹은 플로피디스크로 작업하는것보다 더 많은 품이 든다. Linux등록부구조들이 마치 DOS디스크구동기들이였던것처럼 접근할수 있어야 한다. 이 능력은 체계상에 존재하는 DOS나 Linux구획들 또는 하드구동기들에 들어 있는 자료와 응용프로그람들을 호출하기 위한 DOS환경을 마련한다.

그 처리의 첫 단계는 DOSEmu환경이 구축되기전에 수행된다. 즉 호출하려는 구획들을 Linux등록부구조의 론리적장소로 설치하도록 한다. 실례로 만약 DOS구획이 존재한다면 그것을 /dos로 설치한다.

일단 리용하려는 모든 등록부들과 구획들이 Linux에서 호출가능하면 다음과 같은 지령으로 DOS환경을 마련할수 있다.

$ dos

혹은

$ xdos

일단 DOS가 실행되면 Linux등록부를 DOS등록부문자로 변화시키기 위하여 lredir 지령을 리용한다. Lredir지령의 기본문법은 다음과 같다.

$ lredir *driveletter*: linux\ fs/*linuxdirectory*

이 지령은 규정된 Linux등록부가 제정된 구동기문자를 리용하여 DOS에서 리용될수 있게 한다. 실례로 DOSEmu환경에서 리용하려는 Linux의 /dos등록부를 D구동기로 할당된 DOS구획으로 설치하기 위하여서는 다음의 지령을 리용해야 한다.

$ lredir D: linux\ fs/dos

주의 기정적으로 **lredir**지령은 **c:\ dosemu**등록부에 있다. 만약 이 등록부가 규정된 경로에 없다면 완성된 경로(**C:\ dosemu\ lredir**)를 리용하든가 아니면 **C:\ dosemu**에 경로를 만들어야 한다.

lredir지령의 또 다른 하나의 응용은 리용자들이 리용할수 있는 이미 규정된 DOS구동기에 홈등록부를 만드는것이다. 이를 위해 lredir지령에서 다음의 변수를 리용한다.

$ lredir E: linux\ fs\ $ {HOME}

이 지령에서 리용자의 홈등록부는 Linux환경변수홈으로부터 얻어 진다. 여기에서 Red Hat는 현재 리용자의 홈등록부를 보관한다.

마지막으로 제공된 디스크영상으로부터 초기기동을 하거나 다른 경로에 있는(설치된 DOS구획과 같이) config.sys파일로부터 기동을 한다면 간단한 두개의 단계를 리용할수 있다.

1. DOSEmu환경을 기동하기 위하여 리용하는 초기디스크영상의 config.sys파일을 편집한다. 앞의 실례에서 언급하였지만 그 파일은 /var/lib/dosemu/bootdir.first등록부에 위치하고 있다. 다음과 같이 파일의 첫행을 편집해 놓는다(DOS분구는 Linux의 /dos에 설치되였다고 가정한다.).

   install=c:\ dosemu\ lredir.exe c: linux\ fs/dos

2. 디스크영상에서 config.sys파일을 확인한다. 그러면 DOS구획이 완전히 같다는

것을 알수 있다. 이것은 DOS구획의 config.sys파일에 단계1에서 취급된 지령행이 있다는것을 의미한다.

config.sys와 autoexec.bat를 실행시키기전에는 디스크영상이 설치되지 않으며 DOS부분을 DOS환경 C:로 다시 설정해 주어야 한다.

VMware

VMware(www.vmware.com)은 가상기계기술이다. 그것이 하는 일은 일반적으로 콤퓨터장치들에서 직접 실행하게 될 실제적인 조작체계를 설치할수 있는 완전한 가상콤퓨터를 어느 한 창문에서 실현하여 실행하는 콤퓨터의 모든 자원을 리용하게 하는것이다.

여기서 리용할수 있는 VMware워크스테션판본은 DOS, Windows 3.1, Windows 95/98 그리고 Windows NT/2000, PC의 조작체계들인 FreeBSD, Linux와 같은 공동 Unix배포물들에서 실행할수 있다. Windows XP를 지원하는 방법도 있다. 많은 하드웨어 PC들이 존재하고 그에 설치되여 실행하는 실조작체계들이 각이하므로 호환성은 거의 찾아 볼수 없다. VMware는 Linux체계에서 개발과 검사 그리고 량립성을 위한 다중조작체계를 실행시키는 위력한 방법을 제공한다. 같은 시간에 각이한 조작체계를 실행시켜 VMware의 다중프로그람들을 실행할수도 있다.

물론 위력한 장치는 제공된다. 체계를 합리적으로 구성하자면 pentium II, pentium III, 고속 CPU와 많은 기억기가 필요하다. 왜냐하면 VMware를 가지고 실행하는 Linux와 말단조작체계들은 기억기를 공유해야 하기때문이다. Linux에서 VMware를 실행하자면 최소 96MB의 용량을 가진 기억기가 필요하며 최대로는 256MB이상이 리용된다.

VMware 얻기

VMware는 일반적인 대중공개사용허가의 통제를 받지 않는 업무용프로그람이다. 그럼에도 불구하고 30일동안 무료로 VMware의 표준판본들을 시험할수 있다. 기업Web싸이트 http://www.vmware.com로부터 VMware를 설치할수 있다.

VMware의 소비자판본에는 두가지(VMware Workstation, VMware Express)가 있다. VMware Workstation은 자체 가상기계의 목록화된 조작체계에서 임의의 조작체계를 실행하게 한다. VMware Express는 Linux에서 Windows 95/98만 실행하게 한다. VMware의 판본은 탁상에 의존하며 59～329딸라범위에서 그것을 구입할수 있다. 이 장에서는 Linux의 VMware Workstation 2.0.4를 설치하는 경우를 본다. 만일 여러가지 VMware의 판본이 있다면 수행되는 단계들도 서로 각이하다.

VMware 설치

VMware은 tar.gz파일이나 RPM파일로부터 설치할수 있다. 리용할수 있는 VMware판본에 기초한 일반적인 설치안내서는 모두 같다. 만일 판본이 2.0.4이 아니라 다른 VMware을 구입했다면 리용되는 지령들은 아래에서 보여 준것처럼 변할수 있다. 최근에 나온 http://www.vmware.com Web페지의 문서들을 참고해야 한다.

실례로 /tmp등록부에서 tar.gz파일을 설치했다면 이에 해당되는 파일들을 선택해서 새롭게 설치된 등록부로 이동시키고 뿌리사용자방식으로 가입해서 다음과 같은 지령을 실행시킨다.

```
$ tar xzvf VMvmware-2.0.4-1142.tar.gz
$ cd vmware-distrib
$ ./vmware-install.pl
```

반대로 /tmp등록부에 RPM파일을 설치했다면 이에 해당되는 파일들을 선택해서 새롭게 설치된 등록부로 이동시키고 뿌리사용자방식으로 가입해서 다음과 같은 지령을 실행시킨다.

```
$ rpm -i vmware-2.0.3-799.i386.rpm
$ /usr/bin/vmware-config.pl
```

tar.gz파일이나 RPM파일로부터 기본프로그람을 꺼낸후에 설치안내서를 참고해야 한다. 설치방법은 같다. 일반적으로 VMware의 프로그람들을 실행시킬수 있지만 이미 제29장 《Red Hat Linux 7.1을 Windows와 Novell망에 통합》에서 설명한 Samba프로그람이 설치되여 있다면 다음과 같은 질문에 《no》로 대답하면 된다.

Do you want this script to automatically configure your system to allow your virtual machines to access the host file system?

만약 그렇지 않으면 VMware는 자기식으로 Samba를 구성하게 된다.

주의 만일 VMware의 평가판본을 설치하는중이면 http://www.vmware.com으로부터 평가허가를 적용해야 한다. 콤퓨터의 평가허가설치에 대해서는 전자우편로부터 VMware안내서를 얻어서 참고하면 된다.

VMware에 토대하여 조작체계를 설치

X Window에서 VMware를 실행해야 한다. 먼저 GNOME말단과 같은 지령행대면부 혹은 GNOME이나 KDE의 Konsole과 같은 지령행대면부를 연다. 다음과 같은 지령으로 VMware Workstation을 실행시킨다.

```
$ /usr/bin/vmware
```

Vmware를 실행할 때 그림 30-3에서 보여 준것처럼 3개의 기본선택항목을 가진 차림표가 나타난다. VMware에서 조작체계를 설치하는 가장 간단한 방법은 Run The Configuration Wizard를 선택하고 OK단추를 찰칵하는것이다.

그러면 VMware에는 여러가지 선택이 나타난다. 여기서 다음과 같은것을 선택한다.

Operating System VMware Workstation에서 Windows 3.1, 95, 98, NT 4.0, 2000, Linux 또는 FreeBSD로서 알려 진 전문Unix clone을 설치할수 있다.

Disk Type Setting 새로운 가상디스크 혹은 물리적인 디스크에 요구하는 조작체계를 설정할수 있다.

New Virtual Disk Linux파일에서 가상디스크를 만들게 한다. 이 디스크는 매우 크게 설정할수 있다. 기정적인 최대디스크크기는 2000MB이다.

Existing Physical Disk 이미 같은 콤퓨터에 또 다른 조작체계가 설치되여 있다면 VMware를 통한 호출을 설정할수 있다. 이 부분은 새로운 가상디스크를 설정할 때를 전제로 한다.

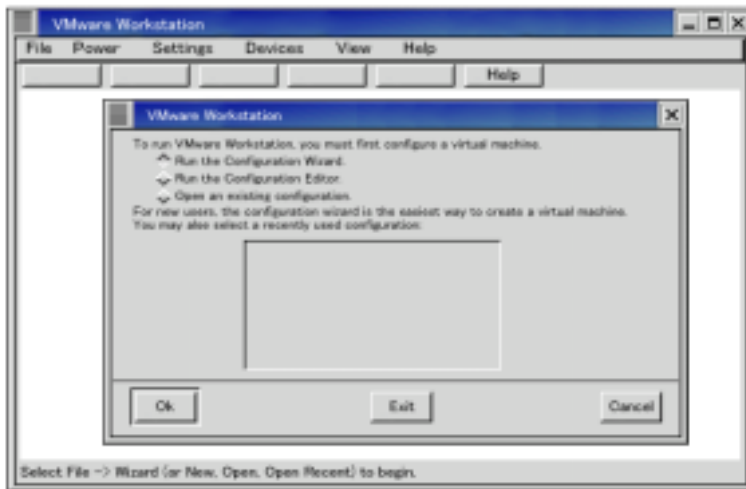



그림 30-3. Vmware의 기동

CD-ROM Access CD-ROM구동기를 호출하기 위하여 가상기억기를 설정할수 있다.

Floppy Access 플로피디스크를 호출하기 위하여 가상기계를 설정할수 있다.

Networking 여기에는 2개의 기본망선택이 있다. 만약 콤퓨터에 Ethernet망기판이 설치되였다면 Bridged Networking을 선택할수 있다. 이것은 현재의 국부망을 호출할수 있게 한다. Host-only Networking선택은 콤퓨터와 가상기계조작체계사이에 두 콤퓨터망을 설정한다.

그것이 끝나면 VMware는 설정된 정보들을 확인한다. 실례로 그림 30-4는 Linux콤퓨터에서 Windows 2000설치에 대한 정보를 보여 준다.

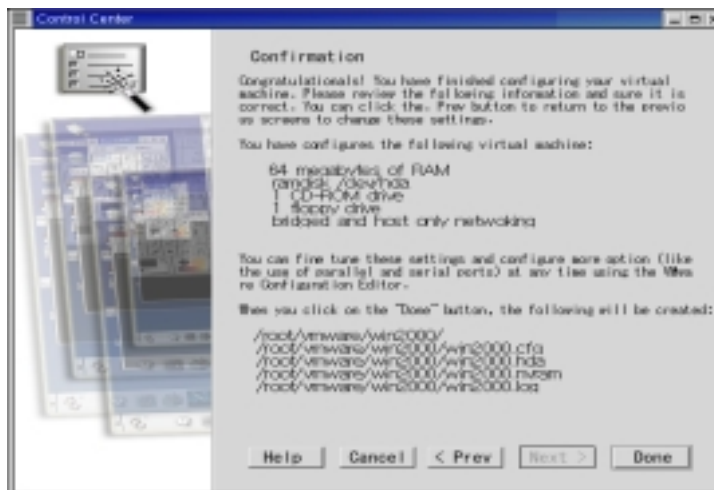



그림 30-4. VMware구성하기

설정을 확인하는 작업이 끝나면 콤퓨터에 가상콤퓨터가 만들어 진다. 빈 VMware화면에서 Power On단추를 찰칵할 때 처음으로 콤퓨터에 전원이 공급된다. 그때 플로피디스크 혹은 CD-ROM구동기의 설치프로그람들을 실행시킬수 있다.

일러두기	Red Hat Linux 7.1은 핵심부판본이 2.4.2이기때문에 판본이 2.0.4보다 작은 VMware프로그람으로는 작업할수 없다.

Linux안에서 Windows 95/98을 실행시키는 또 다른 한가지 업무용선택프로그람묶음은 http://www.netraverse.com에 있는 NeTraverse로부터 리용할수 있는 Win4 Lin이다.

왜 Wine을 리용하지 않는가

Wine은 두가지 원인으로 하여 VMware와 완전히 차이나는 프로그람이다. 첫째 리유는 그것이 업무용프로그람제품이 아니라는것이며 둘째로는 그것이 Linux체계에서 Windows응용프로그람들을 실행하게 하는 완전한 환경을 지원한다는것이다. 여기서 특히 둘째 리유가 중요하다.

현재 Wine의 개발판본은 Windows의 설치에 의존하며 개발목표는 뒤에서 설명하는 둘중의 어느 하나로 규정된다. 만일 Linux체계에 Wine을 설치하였다면 Windows응용프로그람들을 실행시키기 위하여 Windows의 판본을 설치할 필요가 없다.

Linux의 많은 응용프로그람들은 전형적인 원천코드공개프로그람이기때문에 Wine은 끊임없이 계속되는 개발과 수정과정에 있으며 개방성프로그람으로 인기가 높다. 사실상 Wine와 같은 개방성프로그람은 개발코드들을 모두 공개하기때문에 경계를 가지고 리용해야 한다. Wine의 홈페지는 http://www.winehq.org에 있다. 즉 Wine프로그람의 개발속도는 빠르며 새로운 판본은 Web싸이트로부터 쉽게 얻을수 있다.

Wine의 편집물, 구성 및 설치안내서들은 이러한 개발상태를 반영한다. 그러나 Word 6을 비롯한 견고한 Windows응용프로그람들도 Wine에서 실행할수 있다(이 책을 편집하는데서 25%정도는 이 프로그람들로 수행한것이다.).

Wine의 특징

Wine프로그람개발팀은 의의 있는 많은 Wine프로그람개발계획을 가지고 있다. 이 계획들은 다음과 같다.

- Linux, FreeBSD, OpenBSD 그리고 Solaris를 비롯한 기본PC Unix체계에서 Windows 3.1, 95, 98, NT 그리고 2000 을 실행할수 있게 하는것이다.
- Windows를 설치하지 않고도 Windows응용프로그람들을 실행할수 있게 하는것이다. Wine은 코드가 공개된 Microsoft계의 완전한 체계 DLL파일들을 제공할것을 계획하고 있다. 만일 체계 DLL파일들을 가지고 있다면 본래의 Windows체계 DLL파일들을 여전히 리용할수 있지만 Wine이 끝나는 시점에서 그것들은 필요 없다. 계획된 내용들은 다음과 같다.
- 일부분의 DirectX프로그람들은 Windows유희프로그람들을 지원한다.
- 모든 GDI를 지원한다.
- 고유한 Windows 16비트인쇄기구동프로그람들을 지원한다.
- 효과적인 망작업을 지원한다.

- 스캐너들을 지원한다.
- 제한된 범위에서 유니코드를 지원한다.
- 다국어환경을 지원한다.

Wine프로그람은 직책상 개발자들을 위한 프로그람이며 일반적으로는 리용하지 않는다. 그럼에도 불구하고 사람들은 많은 Windows응용프로그람들을 지원하는 Wine프로그람을 비롯한 Linux상에서 Windows응용프로그람들을 실행하기 위하여 이 프로그람을 요구한다.

많은 사용자들은 시험결과를 변화시키면서 Wine에서 수천개의 Microsoft Windows 응용프로그람들을 시험하였다. 흥미를 끄는 응용프로그람들을 다시 보기 위해 Wine자료기지 즉 http://www.winehq.com/Apps/query,cgi를 조사할수 있는 Web페지를 리용해야 한다.

성공한 Wine 프로그람의 연혁

좀 불안하다 할지라도 Wine프로그람은 여전히 성공적이라고 말할수 있다. 다른 software제작자들은 Windows로부터 Linux에로 자기들의 제품을《항구》로 나르는 Wine프로그람의 도구들을 리용하였다. 다른 말로 말하면 그들은 코드를 수정함이 없이 Linux에서 자기들의 프로그람을 실행시킬수 있는 Wine프로그람을 리용한다.

순수한 Word사무처리프로그람을 제작한 Corel회사는 1998년에 Wine프로그람으로 작업하기 시작하였다. 그들은 처음에 판본 7로부터 Linux에서 사무처리프로그람 WordPerfect을 지원하는 Wine프로그람의 판본을 개발하였으며 새로운 판본으로 계속 발전시켜 오늘날에는 Linux의 사무처리프로그람 WordPerfect Office 2000을 개발하였다. 또한 그들은 Wine프로그람을 리용하는 CorelDraw Graphics Suite도 개발하였다.

Corel회사는 Wine프로그람계획실현에 뒤이어 사람들로부터 따로따로 Wine도구들을 개발하였으며 그다음에 그것들을 모두 공유시키고 일반적인 GNU공개사용허가로 보호하였다.

콤퓨터도형설계가중의 한 사람인 Deneba는 Linux에 고회상도를 요구하는 유화그리기응용프로그람을 개발하려고 하였다. 그들은 6달만에 Linux에 유화그리기프로그람 Canvas를 개발하고 그것을 Wine도구로 리용할수 있었다. 사실상 이 프로그람은 Wine문서고들을 약간 수정하는 방법으로 두달동안에 완성하였다.

이 응용프로그람들은 자기자체로 실행할수 있고 VMware와 같은 모의프로그람을 요구하지 않는다. 이것은 Wine이 어떤 일을 하는가를 의미한다. 즉 Wine은 모방기가 아니다.

▶미리 알아두기

이 장에서는 Linux에서 실행할수 있는 Windows와 DOS응용프로그람들의 경로설정방법을 배웠다. 지나간 매주 Linux에서 Windows와 DOS응용프로그람들을 실행할수 있는 기술이 크게 개선되였고 잘 구성된 Linux체계들에서 대부분의 Windows응용프로그람들이 오유없이 실행할수 있을 때까지 시간은 그리 오래 걸리지 않았다.

제31장 《보안과 값 눅은 경로기로서의 Red Hat Linux 7.1》은 흥미 있는 문제들

을 보여 준다. 든든한 망조작체계로서 Linux는 소규모망에서 경로기로 직접 작업할수 있고 모뎀이나 IDSN을 통하여 인터네트에 접속할수도 있고 내부LAN경로기로서 다른 망에 련결할수도 있다.

특히 이 장에서는 소규모사무작업을 위한 전형적인 실례 즉 하나의 IP주소만을 지원하는 전화망접속으로서 인터네트에 LAN을 접속하는 방법을 고찰하게 된다.

제 31 장. 보안과 값 눅은 경로기로서의 Red Hat Linux 7.1

이 장에서는 서로 련관되여 있는 보안과 인터네트의 망접속문제들에 대하여 보게 된다.

여기에서는 여러 단계에서 Linux보안을 고찰한다. 우선 개별적인 Linux체계들을 어떻게 보안하는가를 보며 다음으로 인터네트와 접속할 때 LAN보안을 어떻게 하는가 하는 좀더 넓은 범위의 문제를 고찰한다. 또한 체계를 설치할 때나 혹은 설치한후에 그것을 보안하기 위한 방화벽구성법에 대하여 본다.

더 나아가서 Linux경로기설정과정을 보게 되는데 여기서는 특유한 Linux배포물을 리용하여 망과 인터네트사이를 접속하는 관문으로서의 Linux경로기를 구성한다.

기초적인 보안문제

콤퓨터보안은 책의 단 하나의 장으로 다 취급하기에는 그 범위가 대단히 넓다. 완전하다고 하는 책들도 실지로는 콤퓨터의 작은 범위들과 망보안에만 국한되여 있다.

문제의 범위가 주어 진 조건에서 이 절에서는 일반적인 콤퓨터보안에 대하여 설명하며 특유한 세부들에 적용할수 있는 기본개념들을 고찰한다.

독립실행가능한(Stand-Alone) 체계보안

독립실행가능한 Linux체계에는 가장 기초적인 보안요구들이 있다. 모뎀을 통하여 인터네트에 접속되였다고 할 때 체계가 LAN에 접속되거나 오래동안 접속되지는 않으므로 체계보안문제는 비교적 간단해 진다.

아직도 몇가지 기초적인 보안문제를 취급해야 한다.

- 암호문은 매우 복잡하여 쉽게 해석할수 없거나 해커들로부터 쉽게 공격 받을수 없도록 한다. 암호문은 문자결합, 수자, 임의의 비단어문자들로 구성한다. 또한 암호문은 자주 변화시킬수 있다. 특히 중요한것은 뿌리암호문이 비단어이며 정기적으로 변화시킬수 있는 복잡한 암호문이라는것이다.

주의 암호문을 해석하기 위한 특수한 프로그람들이 많이 만들어 졌다. 사전단어에 기초한 암호문은 현재의 PC를 리용하면 순간에 해석할수 있고 수자들과 대소문자들로 혼합된 암호문은 그를 해석하는데 같은 개인용콤퓨터로서 몇주일 걸린다. 복잡한 암호문을 푸는 한가지 쉬운 방법은 적당한 문법규칙을 적용하는것이다. 실례로 OM151ftp과 같은 복잡한 암호문은 "On March 15,I'm flying to Paris."로부터 만들수 있다.

- 콤퓨터가 장치적으로 보안기능을 가지고 있는가를 확인한다. 만일 콤퓨터에 중요한 자료가 있으면 점심식사시간에도 사무실을 비우지 않는다. 이와 마찬가지로 조종탁 혹은 말단창문에서 뿌리용자로 가입하였을 때 사무실을 비우거나 체계가입을 그만두어서는 더욱 안된다.
- 불필요한 사용자에게는 권한을 주지 않는다. 규정에 기초하여 오직 두 사람만이 체계를 리용한다면 이 사용자들에게만 권한을 주고 그밖의 사람에게는 권한을 주지 않는다. 이러한것은 누가 체계를 사용하고 있는가와 어디서 어떤 보안규정을 위반하고 있는가를 쉽게 알아 낼수 있게 한다.
- 그중에서 가장 중요한것은 규정에 기초하여 체계를 리용할수 있다고 인증된 사람들도 암호문(사용자 혹은 뿌리사용자)을 공유하지 않는다는것이다. 특히 뿌리사용자암호문을 잘 보관하여야 한다. 뿌리암호문을 잘 보관하지 못하면 뿌리암호문이 불법가입자에게 리용될수 있고 결국 체계파괴로 자료를 잊어 버리는 후과를 빚어 내게 될것이다.
- 체계는 완전히 보안할수 없기때문에 중요한 자료들은 암호화하여야 한다. 이렇게 하는데는 PGP암호화를 비롯하여 많은 도구들이 있다. PGP를 포함하는 HOWTO 는 http://www.linuxdoc.org/HOWTO /Mutt-GnuPG-PGP_HOWTO. html에서 직접 리용할수 있다. PGP암호화에 대한 구체적인 정보는 또한 인터네트의 LinuxFocus Web싸이트주소 http://www.linuxfocu s.org/English/November1997/ article7.html에서 찾을수 있다.

LAN 에서의 Linux 체계의 보안

LAN에서 Linux체계를 즉시 접속하면 새로운 보안문제가 제기된다. 독립실행가능한 체계에서 앞서 소개한 모든 보안규칙들을 적용했다 하더라도(체계가 LAN우에 있을때 가능하다.) 해커는 망을 통하여 극히 적은 보안약점을 공격하는 방법으로 Linux체계를 호출한다.

이러한 문제는 두가지 간단한 방법으로 극복할수 있다.

- 불필요한 망봉사기능을 실행시키지 말아야 한다. 실례로 사용자들이 전화망(telnet)으로 체계에 접속하지 않도록 한다면 전화망데몬(/usr/sbin/in.telnetd)이 설치되지 않았는가, /etc/xinetd.d등록부안에 있는 데몬입구가 없는가를 확인한다. 이와 류사하게 콤퓨터가 우편봉사기로 동작하지 않는다면 sendmail데몬을 해제한다. 이와 같은 방법을 대부분의 망봉사기들 즉 FTP, finger, news,DNS와 그밖의 다른 많은 망봉사기들에 적용한다. 특별히 망에 접속할 필요가 있는 망봉사기데몬들만 기동하고 Linux체계의 과제를 수행한다.

일러두기 만일 Linux의 다른 판본들(Red Hat Linux 6.x을 포함하여)에서 **/etc/inetd.conf**파일을 공동으로 사용한다면 그 파일은 **/usr/sbin/inetdconvert**지령을 가지고 Red Hat Linux 7.1의 **xinetd** 양식으로 변환할수 있다. 만일 **/etc/xinetd.conf**파일이 없는 Linux의 다른 판본을 사용한다면 **/etc/xinetd.conf**파일안의 데몬질문에 대하여 입구가 없다는것을 확인한다(데몬과 지령의 앞에 하쉬표식[#]을 첨가하는것으로 설명문을 달수 있다.).

- 제28장 《이써네트망을 위한 Red Hat Linux 7.1 구성》에서 론의된 /etc/hosts.allow 와 /etc/hosts.deny파일들을 효과적으로 리용하자. 모든 콤퓨터들을 그 이전처럼 동작할수 없게 하기 위하여 IP주소로부터 호출되는 모든 봉사를 막는다. 뿐만아니라 그 봉사를 받게끔 허가된 콤퓨터들은 망자원을 호출할수 있도록 한다. 실례로 오직 두 사용자만이 FTP봉사기를 호출할수 있도록 설정하면 바로 그 사용자들만 FTP봉사기를 호출하도록 하고 임의의 LAN사용자에 대해서는 FTP접속을 허가하지 않는다.

xinetd

Red Hat Linux 7.0로 시동한다면 Red Hat는 inted의 재배치로서 확장된 인터네트봉사데몬 혹은 xinetd기능을 추가한다. 인터네트를 조종하는 고성능봉사기 inetd는 포구로 Unix/Linux콤퓨터들을 호출한다. Telnet, FTP, POP전자우편과 같은 봉사들을 경험이 부족한 사용자들이 빨리 사용할수 있게 하자면 기정으로 inetd을 열린 포구들로 설정하여야 한다. 우에서 서술한바와 같이 콤퓨터에 정상적으로 가입하려면 hosts.allow와 hosts.deny파일을 리용해야 한다. 만일 하나의 콤퓨터가 FTP를 내리적재한다면 콤퓨터우에서 모든 봉사를 받을수 있다.

xinetd데몬은 인터네트우의 모든 봉사를 정기적으로 받을수 있게 한다.

다시말하여 모든 사람들이 FTP봉사를 호출한다면 Telnet나 NFS와 같은 다른 봉사들은 호출하지 않도록 해야 한다. 만일 그렇게 하지 않으면 열려 진 FTP봉사기에 위험이 조성된다.

xinetd의 다른 우월성은 봉사거절(DoS:Denial of Service)이라는 기능을 리용하여 침습해 들어 오는 해커들로부터 콤퓨터를 보호할수 있다는것이다. DOS는 봉사기에 끊임없는 통보문을 흘러 보내는 처리로서 봉사기를 호출하는 콤퓨터들을 보호해 준다.

xinetd에 대한 보다 많은 정보를 보려면 http://www.xinetd.org의 Web싸이트를 참고해야 한다. 이 싸이트에는 필요한 구성파일의 문서와 실례가 있다.

망보안의 보존

LAN에서 Linux체계보안을 완전히 보존한다 할지라도 LAN이 인터네트에 접속되면 망은 해커들로부터 위험과 공격을 받게 된다.

망보안을 보존하는것은 매우 중요한 일이며 이를 위해서는 망보안에 대한 깊은 지식이 있어야 한다. 망보안을 안전하게 보존하는데는 최소한 두가지 기술(그리고 그의 많은 변종들)이 있다.

LAN을 절환하는 방법으로 망이 자료전송경로기를 통해서만 인터네트에 접속된다는 것을 잊지 말아야 한다. 일반적으로 경로기는 해커들이 LAN을 침범할수 없도록 하는 몇가지 보안기술을 가지고 있다.

경로기보호는 보통 두가지 방법중 하나를 리용한다.

파케트려파방화벽의 리용 파케트려파(Packet-filtering)는 매 경로기들에 다 있다. 경로기는 인터네트에 정보파케트를 수신하고 LAN의 체계정보와 IP시작주소, IP목적지주소, 접속포구들을 규정할 때마다 파케트를 LAN에서 전송하거나 망에서 소실되는것을 결정하는 표로 검사하고 비교한다. 이런 방법으로 콤퓨터접속 및 특별한 포구들(혹은 이

것들의 임의의 결합)이 허가되거나 거절될수 있다.

대용방화벽의 리용 대용방화벽(proxy firewall)은 방화벽내부의 장치들과 인터네트장치들사이의 직접적인 접속을 조장하는 독특한 입구이다. LAN의 주콤퓨터와 인터네트의 주콤퓨터가 접속한다고 할 때 실지 그것은 앞뒤로 전송하는 모든 자료를 봉사하면서 대용체계와 접속하며 그 대용체계는 또 인터네트체계와 접속한다. 이와 비슷하게 만일 LAN과 임의로 가입접속하면 대용체계는 중개자와 같은 역할을 한다. LAN과 인터네트의 차이를 무시하면 대용방화벽은 려파프로그람파케트방화벽과 비슷한 능력을 가지고 있다는것을 알수 있다. 대용방화벽은 IP시작주소와 IP목적지주소, 사용포구를 포함하는 여러개의 기초변수들을 가지고 접속을 거절하거나 승인하는 방화벽을 통하여 확증된 통화형태만을 전송하도록 구성한다.

망보안의 기초적인 개념을 리해하려면 http://www.linuxdoc.org/ HOWTO/ Security-HOWTO.html에서 Linux Security-HOWTO를 참고해야 한다.

Linux체계는 LAN과 인터네트를 련결하는 경로기로서 쉽게 구성할수 있고 전화모뎀, 케블모뎀 혹은 DSL기판과 같은 고속접속장치들을 통하여 표준접속을 진행할수 있다. 이 Linux경로기는 려파프로그람파케트 혹은 대용방화벽처럼 잘 작용하도록 구성할수 있다. 일반적으로 많은 장치들로 구성된 경로기보다 더 많은 보안기능을 제공해 주고 원가도 적은 Linux경로기를 설정한다.

Linux 설치기간 방화벽지원구성

콤퓨터를 다른 망에 접속할수 없다면 인터네트에 련결할수 없으며 방화벽도 필요없다. 만일 짧은 시간동안 인터네트와 접속한다면 그만큼 망이 침범당할 위험도 적어 지고 방화벽설정기능도 간단해 진다. 그러나 콤퓨터가 인터네트에 자주 접속되는 망에 접속했다면 모든 종류의 공격들을 다 받을수 있다. 콤퓨터가 DSL이나 케블모뎀과 같은 “항상 련결”방식으로 인터네트와 직접 접속되였다면 가장 큰 위험을 당할수 있다.

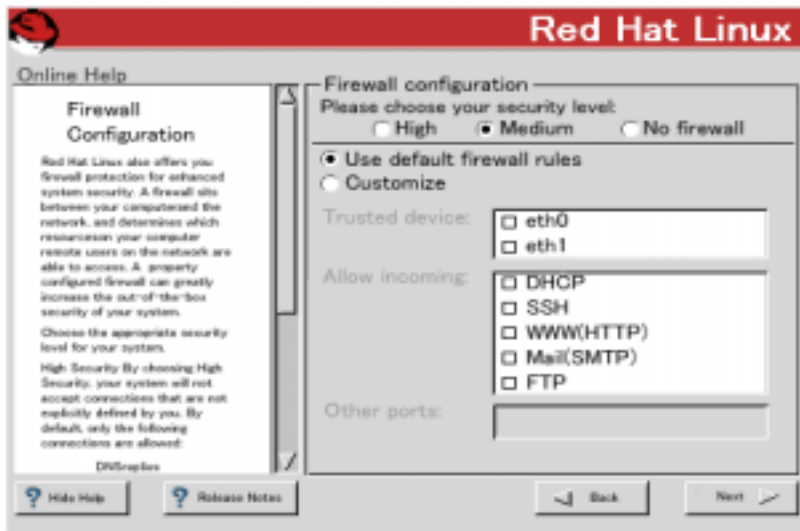



그림 31-1. Linux설치기간에 방화벽구성

그림 31-1에서 보여 주는바와 같이 Red Hat Linux 7.1설치과정에 직접 Linux체계에 방화벽을 구성할수 있다. 새 방화벽구성을 설정할수 있는 lokkit도구도 준비되여 있다. 끝으로 간단한 방화벽을 해석하고 그 위력에 대한 리해를 얻을수 있다.

그외에도 방화벽은 일반적으로 모든 정보통신량을 블로크화하여 전송한다. 이것은 방화벽이 규정된 포구나 통로를 통해서 확정된 자료만을 전송하는 전용포구와 같다고 말할수 있다.

인터네트규약 TCP/IP에는 총 65,536개의 포구들이 있다. 이 매 포구들은 자료전송을 위한 전용포구들과 같다. 콤퓨터들은 기초통신을 위해 일부 개방된 포구들을 요구한다. Red Hat Linux 7.1설치프로그람은 고급 혹은 중간급보안방화벽을 설정하는데 이러한 방화벽은 개방된 전용포구라고 말할수 있다.

주의 65,536개의 TCP/IP포구들은 세개 부분으로 나눈다. 잘 알려 진 0부터 1023까지의 포구들은 Web페지, 전자우편 그리고 인터네트대화통로들과 같은 봉사기능들이며 1024부터 49,151까지 등록된 포구들은 인터네트전화와 NFS와 같은 봉사기능을 제공한다. “ad hoc”에서는49,152부터 65,535까지의 포구들을 리용할수 있다. 관리목록은 www.iana.org/assignmemts/port-numbersd에서 수자권한으로 할당된 인터네트를 통하여 리용할수 있다. 이 많은 포구들은 /etc/services파일에 들어 있다.

믿음직한 장치

체계를 설치할 때 검출된 모든 망기판들은 방화벽구성화면에 표시된다. 그림 31-1은 두개의 망기판 즉 eth0과 eth1들을 보여 준다. 이 콤퓨터가 eth0은 국부망에 련결되고 eth1은 인터네트에 련결된 두 망사이에 있다고 하자. 이 기판들중 하나를 택하면 그 장치에 방화벽을 설치할수 있다.

국부망에 있는 모든 사용자들의 콤퓨터들을 선정하려면 Customize 선택항목을 선택한다. 이것은 eth0을 선정된 장치로 등록한다.

다른 한편 인터네트에로의 그 어떤 접속도 설정하지 말아야 한다. 이 경우에는 eth1을 선택한다. 그렇지 않으면 방화벽은 인터네트를 통한 해커들의 공격을 막지 못할것이다.

고급보안

고급보안방화벽은 바깥망과는 전혀 접속을 가질수 없다. 이런 경우에 대부분의 사용자들이 최소한의 인터네트접속을 바란다는것은 일반적으로 무의미하다. 이런 경우에 가장 좋은 한가지 방법은 하나의 방법만으로 접속되는 방화벽을 리용하는것이다. 즉 자료는 밖으로 전송될수 있지만 방화벽은 들어오는 모든 자료들을 블로크화한다. 리론적으로는 대단한것이며 때문에 이런 방화벽을 설치하면 인터네트와 수시로 접속할수 있다.

Web페지의 도움을 받으면 인차 알수 있지만 기정 Red Hat Linux 7.1의 고급보안방화벽은 두가지 종류의 정보 즉 DNS봉사기로부터 받을수 있는 콤퓨터주소정보와 DHCP봉사기로부터 받을수 있는 구성정보를 리용한다.

DNS 제21장《Linux망에 대한 리해》에서 본바와 같이 령역이름봉사기(DNS)로 알려 진 이름봉사기는 www.sybex.com와 같은 령역이름들이 있는 Web자료기지이고 그에 해당한 IP는 192.168.0.131와 같이 주소화되여 있다. 콤퓨터는 인터네트와 대화할 때 요구하는 Web페지를 인차 찾을수 있도록 IP 를 주소화한다. 만일 설치파정에 어떤 DNS봉사기의 IP주소를 (/etc/resolv.conf 파일을 검사한다.) 얻는다면 방화벽편의프로그람은 포구53을 열어 준다.

DHCP 제4장《Red Hat Linux 7.1설치》에서 본바와 같이 동적주콤퓨터구성규약(DHCP)봉사기는 국부망에서 IP 주소정보를 제공한다. 만일 망밖에서 DHCP봉사기를 사용한다면 봉사기는 방화벽을 통하여 콤퓨터와 대화를 할수 있다. 만일 그림 31-1에서 Allow Incoming구간의 DHCP를 선택한다면 방화벽편의프로그람은 포구67과 68을 열어 준다.

고급보안방화벽은 다른 정보통신량들은 취급하지 않는다. 이 경우 RealAudio 혹은 대부분의 대화(chat) 프로그람들과 같은 일반 인터네트도구들은 리용하지 않는다. 이런 방화벽을 통해서는 NFS파일을 공유할수 없다. 만일 콤퓨터에 Web봉사기를 가지고 있다면 누구나 Web페지를 리용할수 있다.

중간급보안

중간급보안은 방화벽의 보다 더 많은 열린 포구들을 설정한다. 잘 알려 진 포구들(0부터 1023까지)뿐만아니라 NFS(2049), X봉사기(6000-6009), X폰트(7100)도 닫아 버린다. 앞부분에서 서술한바와 같이 열려 진 포구들은 DNS 와DHCP를 사용하도록 한다. 다시 말하여 중간급보안은 등록포구들을 그대로 열어 놓는다. 이것은 RealAudio 방송망과 같은 규정된 봉사들을 자동적으로 받게 한다.

만일 중간급보안방화벽안에서 열려 진채로 놔둔 포구들을 사용하고 있다면 고급보안을 선택한다. 그리고 전용선택에 대한 다음 부분을 읽는다. 봉사들을 알고 있다면 전용선택을 리용하여 필요한 포구들을 곧 열수 있다.

전용방화벽설정

만일 고급 혹은 중간급보안방화벽이 너무 큰 제한을 받는다면 선택한 포구들을 해제하는 설정을 할수 있다. 그림 31-1에서 보여 준 화면에서 Customize선택항목을 선택한다. 이것은 방화벽의 포구들을 열어 필요한 자료를 준다. Allow Incoming목록은 인터네트접속에 요구되는 대표적 정보통신량들을 보여 준다. Other Ports본문칸은 다른 포구들을 선택하여 열어 준다.

통화량수신

Allow Incoming목록에는 6개의 대표적 포구들이 있다. 이 포구들은 인터네트와 접속할 때나 다른 사용자들이 Web,전자우편,FTP봉사기로 콤퓨터와 접속할 때 요구하는 자료와 관련된다.

DHCP Red Hat Linux설치때 기정으로 선택된다. 앞절에서 설명한것처럼 그 목적은 망에 콤퓨터를 구성하기 위하여 망밖에서 DHCP봉사를 하자는데 있다.

SSH 보안쉘에 대하여 간단히 보자. SSH를 사용할 때 전체 가입대화조종은 암호화된다. 만일 보안쉘프로그람묶음을 리용하여 먼 곳에 있는 콤퓨터를 관리하려 한다면 설정한 방화벽을 통하여 접속하여야 한다. 허가된 SSH는 보안쉘에 가입하게 하는 포구22를 연다. SSH를 사용하려면 openssh-sever-2.5.2p2-5.i386.rpm프로그람을 리용해야 한다.

Telnet Telnet는 먼 곳에 있는 콤퓨터에서 지령들을 등록하고 기입하기에는 불안전한 방식이다. 일단 Telnet가 먼 곳의 콤퓨터를 련결하면 마치도 국부적으로 련결된것처럼 즉시에 지령들을 사용할수 있다. Telnet통신이 암호화되지 않았을 때 망도청기를 가지고 있는 사용자는 (이 도구는 *sniffer*로 알려 져 있다.) 망에서 리용하는 암호나 다른 정보들을 기록할수 있다.

WWW(HTTP) 콤퓨터에서 Web봉사기를 선택하면 이 선택권이 허가된다. 한편 콤퓨터로부터 Web페지를 얻으려는 사람들은 아무것도 찾을수 없다. 이 선택권은 인터네트에서 Web페지를 얻는데 리용되지 않는다.

Mail (SMTP) 이 선택권은 콤퓨터에 우편봉사기가 있을 때 가능하다. 전자우편을 보내려는 사람들은 우편봉사기로 통보를 송신할수 있다. 만일 Mail포구(25)가 열리지 않는다면 전자우편은 방화벽을 통과할수 없다.

FIP 이 선택권은 콤퓨터에 FTP봉사기가 있을 때 가능하다. 공동 FTP봉사프로그람은 이 책에 따라 들어 온 CD-ROM에 wu-ftpd-2.6.1-16.i386.rpm파일로써 보관되여 있다.

포구통화량전용화

Allow Incoming 구간에 렬거된 포구들은 모든 사용자들을 만족시키지 못한다. 실례로 방화벽을 통하여 다른 Linux,Unix콤퓨터들과 등록부를 공유하려면 포구배치나 NFS 출입구들을 다 열어야 한다. 이것은 포구 111과 2049에 해당된다

다음 단계에서는 매 포구들을 거쳐 정보통화량의 부류를 본다. /etc/services파일이나 그 전에 언급된 Web페지를 연다. 이에 상응한 정보는 다음과 같다.

```
sunrpc   111/tcp    portmapper  #RPC 4.0  portmapper  TCP
sunrpc   111/udp    portmapper  #RPC 4.0  portmapper  UDP
nfs      2049/tcp   nfsd
nfs      2049/udp   nfsd
```

이 정보에 기초하여 그림 31-1에서 보여 준 방화벽구성화면에 상응한 출입구들을 열수 있다. Customize를 찰칵하고 Other Ports본문칸에 다음과 같이 써넣으시오.

```
111: tcp,111:udp,2049:tcp,2049:udp
```

이것은 포구 111과 2049를 여는데 이 두 포구를 리용하여 NFS를 통한 TCP 와 UDP정보통화량들을 공유한다. 같은 방법으로 다른 포구들을 첨가할수 있다. 반점으로 포구들을 구분하며 입구량들사이에 공백이 있으면 안된다.

설치후 방화벽지원구성

Red Hat Linux 7.1을 설치한 후에 lokkit 로 방화벽을 재구성할수 있다. 필요하다면 /mnt/cdrom등록부에 CD-ROM 을 설치하고 다음의 지령으로 이 프로그람묶음을 설치한다.

```
# rpm  -i /mnt/cdrom/RedHat/RPMS/lokkit-0.43-6.i386.rpm
```

일단 설치하면 대면부에서 다음과 같은 지령으로 lokkit 를 시동할수 있다.

```
# /usr/sbin/lokkit
```

lokkit를 열 때 그림 31-2에서 보여 준 방화벽구성화면이 나타나는데 이것은 고급 및 중간급보안을 설정하거나 방화벽을 다 없애 버린다. lokkit 에서 Tab건과 공백권으로 선택항목들을 선택할수 있다.

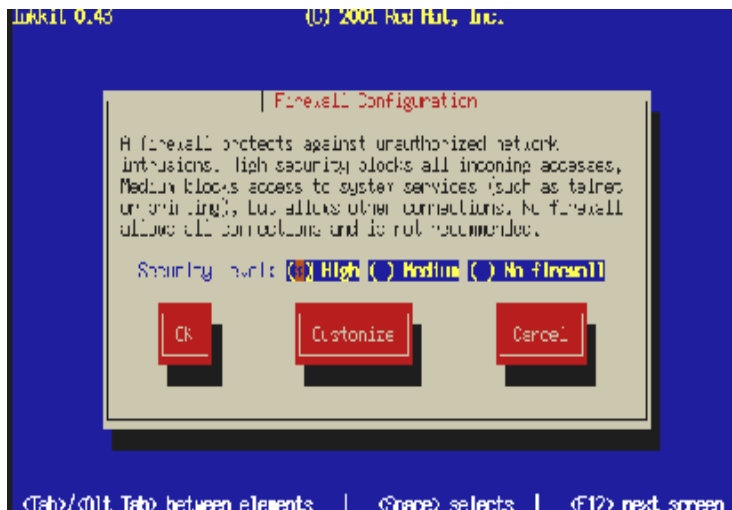



그림 31-2. lokkit방화벽구성화면

고급 혹은 중간급보안을 선택하고 Customize 를 찰칵한다. 그림 31-3에서 보다싶이 Firewall Configuration-Customize차림표가 나타난다.

방화벽구성차림표에 Red Hat설치때 보았던것과 같은 전용선택항목이 나타난다. 이때 망기판선택항목들은 적외선장치(irlan0)와 표준이써네트기판 (eth0)을 포함한다. 이미 서술된 방법을 리용하는 선택권을 확인하고 OK를 찰칵한다. 다음 그림 31-2에서 보여 주는 원래 화면에로 돌아 온다. OK를 찰칵하면 lokkit문서는 /etc/sysconfig /ipchains파일에 새로운 방화벽구성정보를 설정한다.

주의 설치후에 lokkit를 리용하면 기정적인 고급 및 중간급보안방화벽은 DHCP포구를 열지 않는다.

주의 Red Hat Linux 7.1에서 기정적인 방화벽지령은 ipchains이다. 2.4.x핵심부

에서의 새로운 방화벽지령은 iptables이다. ipchains 를 비능동으로 할수 있으며 그때 iptables는 자동적으로 방화벽에서 사용불가능으로 된다. 그러나 lokkit편의프로그람은 iptables 의 구성을 지원하지 않는다. iptables에 대한 서술은 아직 끝나지 않았다. HOWTO는 http://va.samba.org /netfilter/unreliable-guides/packet-filtering-HOWTO /index.html 에서 찾을수 있다. 이것은 www.linuxdoc.orgWeb싸이트에서 찾을수 있다.

주의 두번째 방화벽구성편의프로그람 /usr/sbin/firewall-config도 사용할수 있다. 어쨌든 그것은 lokkit보다는 사용하기 더 힘들다.

그림 31-3. 전용방화벽

IP주소로 여는 포구들을 전용화할수 있다. 다시말하여 특수한 IP주소를 가진 콤퓨터만이 특수한 자료를 보내는 포구를 열수 있다. 이것을 설치하는 수법은 이 책에서는 설명하지 않는다. 방화벽과 망구성에 대한 자료를 더 보려면 로보트제글로가 쓴 *Linux Firewalls* (New Riders, 1999)을 참고해야 한다.

다음절에서 플로피디스크에 기초한 Linux판본에서 방화벽을 구성한다. 여기에는 위장된 정보가 있는데 그 정보는 lokkit에 기초한 방화벽과 결합하여 사용할수 있다. 망내부에서 콤퓨터들의 주소들을 숨기는것으로 위장한다.

Linux에 기초한 경로기의 만들기

이 절에서는 Linux의 Linux경로기대상과제의 배포물을 리용하여 Linux에 기초한 경로기를 창조하는 단계들을 보게 된다.

소규모적인 Linux설치로ㄱ 설계된 Linux의 Linux경로기대상과제의 배포물은 대부분의 싸이트들에 대한 단일한 1.44MB 플로피디스크에 적합하며 두개의 이써네트 LAN

들사이 혹은 이써네트 LAN과 PPP를 통한 인터네트사이의 경로선택에 필요한 모든 구성요소들을 제공한다. 모듈들을 추가하여 WAN접속을 위한 보충적인 하드웨어를 지원할수 있다. SNMP관리와 같은 다른 특성들도 사용할수 있다.

Red Hat와 같은 Linux 의 완전판본대신에 Linux경로기대상과제의 Linux배포물을 리용하여 경로기를 설치하는것은 모든 하드작업을 우리가 진행한다는것으로 하여 매우 편리하다. 작업에 필요한 경로선택을 하는데 무엇이 필요한가를 알 필요는 없다. 보충적으로 쏘프트웨어는 구성정보들을 요구하는 모든 설정작업을 검사하도록 간단한 차림표구동형 구성체계를 포함한다.

Linux경로기대상과제쏘프트웨어는 http://www.linuxrouter.org/에 있는Web싸이트로부터 내리적재할수 있다.

다음의 절들은 보안 및 비보안망들사이에 경로기를 어떻게 구성하는가에 대한 표준실례를 보여 준다. 공교롭게도 Linux경로기대상과제(LRP)쏘프트웨어는 Linux핵심부 2.0과 2.2에 기초한다. 핵심부 2.0에 대한 LRP쏘프트웨어는 보안구멍을 가지는 낡은 ipfwadm지령을 포함한다. 핵심부 2.2에 대한 쏘프트웨어는 ipchains지령을 포함하는데 이것이 더 안전하다. 아직까지 Linux핵심부 2.4와 관련한 iptables지령과 대비할만한 완성된 준위에서의 지원은 없다.

무엇을 만들겠는가

방화벽으로서 봉사하는 경로기를 만들려면 인터네트로부터 LAN비공개구역을 보호하면서 Linux경로기대상과제쏘프트웨어를 리용하여야 한다. 이것은 다음과 같은 문제들을 전제로 한다.

- 인터네트에 접속되여 있다면 케블모뎀이나 DSL기판과 현존경로기를 통하여 혹은 ISDN접속이나 ISDN경로기를 통하여 하여야 한다.
- 망에 있는 봉사기들에 가입호출을 요구한다. 실례로 공동으로 호출할수 있는 Web봉사기가 있어야 한다. 그러나 불안정한 파일과 응용프로그람, 망거처지의 자료기지봉사기들은 호출하지 말아야 한다.

주의 케블모뎀,DSL,혹은 ISDN의 많은 봉사자들은 자기의 전용콤퓨터들이 공개 Web, FTP,혹은 다른 형태의 봉사기들이 같은 준위에서 실행하지 못하도록 한다. 의문나는것은 봉사자에게 물어 본다.

이러한 전제들로 하여 망구조는 그림 31-4와 비슷할것이다. 창조한 내부경로기는 보안내부망과 호출가능한 비보안망사이에 놓이게 된다.

외부세계와 접속한 경로기는 비보안망 및 보안망들사이의 경로를 설정하는 경로기보다 더 쉽게 보호한다는것을 알아야 한다. 이것은 최악의 경우에 오직 비보안망에 있는 체계만이 외부로부터 직접 해커들의 공격을 받을 가능성이 있다는것을 의미한다. 때문에 비보안 및 보안LAN들사이에 있는 두번째 경로기로 해커들의 공격을 막고 보안망에로 가입하지 못하게 하는 안전한 망환경을 구성하기는 힘들다. 비보안 및 보안 LAN들사이

에서의 이 두번째 경로기는 Linux경로기대상과제쏘프트웨어로 실행하여야 한다.

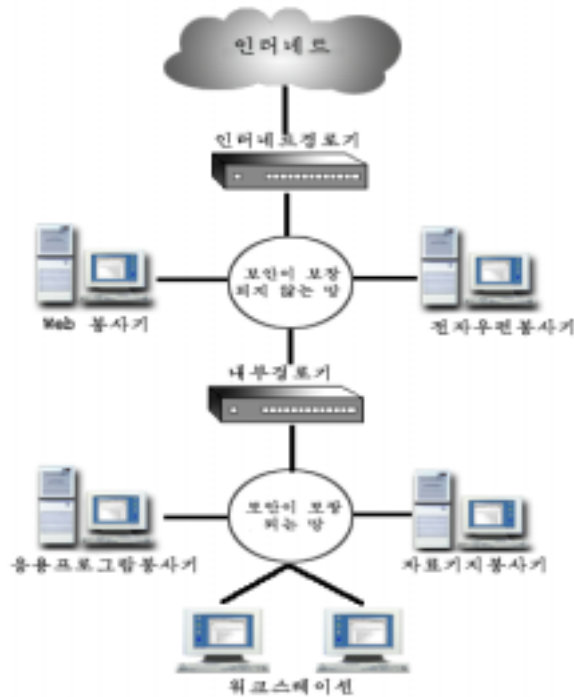



그림 31-4. 대표적인 보호방화벽망

준비

다음으로 현행Linux체계에 의한 Linux경로기플로피디스크를 만든다. 앞에서 설명한것만 가지고도 http://www.linuxrouter.org/down load.shtml Web페지에 목록화된 싸이트들중에서 어느 하나로 이 파일들을 내리적재할수 있다.

1. Linux “idiot-image”파일을 내리적재하고 알맞는 장소에 보관한다. 실례로 /tmp/에 보관한다. 이와 같이 현행 “idiot-image”파일은 idiot-image_1440KB _FAT_ 2.9.8_Linux_2.2.gz이다.
2. cd /tmp와 같은 지령으로 파일을 내리적재한 등록부로 현행등록부를 변화시킨다.
3. 문서를 압축한다. 프로그람들은 gzip형식으로 들어 오며 이 gzip파일들은 다음의 지령으로 풀수 있다.

   gunzip idiot-image_1440KB_FAT_2.9.8_Linux_2.2.gz

4. 플로피디스크에 디스크영상파일을 복사한다. 이렇게 하면 디스크의 임의의 자료는 지워진다. dd지령을 리용하자.

   dd if=idiot-image_1440KB_FAT_2.9.8_Linux_2.2 of=/dev/fd0

이렇게 하면 Linux경로기대상과제배포물에 대한 초기시동플로피디스크를 가지게 된

다. 그러나 이것은 원만한것이 못된다. 이런 때에는 같은 핵심부판본 2.2.16-1.을 리용한 망기판구동기가 필요하다.

2.2.16-1.tar.gz파일로내리적재한 같은 싸이트에서 유용한 기본처리를 반복한다. 이 파일을 /tmp등록부에 복사한다. tar xzvf 2.2.16-1/modules/net지령으로는 이 파일을 열거나 보관할수 없다. 망모듈을 보관하지 않은 /tmp/2.2.16-1/modules/net보조등록부에서 두번째 플로피디스크에로 복사한다. 후에 사용하기 위하여 두 플로피디스크를 보관한다.

주의 이와 같이, Linux경로기대상과제판본 2.9.8은 “beta quality”이다. 다시 말하여 자기의 사명을 다 할 때까지 이 제품의 용법에 대하여 언급하지 않는다.

경로기를 기동하려면 충분한 하드웨어가 있어야 한다. 경로기기능을 수행하는 8MB의 RAM을 가진 386SX와 12MB의 RAM을 가진 486DX266MHz가 상당하다고 해도 Pentium 100MHz나 적어도 16MB의 RAM을 가진 더 빠른 체계로 시동하려고 할것이다. 하드구동을 요구함이 없이 곧 플로피구동기로 초기시동해 보자. 만일 Linux경로기구성에 add-ons 기능을 많이 첨가하면 쏘프트웨어가 플로피디스크의 크기를 초과할수 있다. 이 경우에는 하드디스크나 초기시동가능한 Zip구동기로부터 초기시동한다는것과 RAM이 32MB로 증가한다는것을 고찰한다. 다른 매체로부터 어떻게 초기화하는가에 대한 구체적인 내용은 Linux경로기대상과제홈페지에서 찾을수 있다.

경고 만일 인텔 386이나 486을 장비한 콤퓨터를 사용한다면 2.0.36핵심부에 기초한 낡은 ipfwadm지령들을 가진 “idiot-images”선택항목이 제일 유용하게 된다. idiot-image_1440KB_FAT_2.9.8_Linux2.0.gz image파일과 2.0.36pre15-1.tar.gz 핵심부모듈파일을 리용하여 우의 단계들을 반복한다.

주의 Slackware에는 Zip구동기를 설치하기 위하여 설계된 다른 Linux배포물이 있다. 더 많은 정보는 http://www.Slackware.com/zipslack를 보면 된다.

PC에는 두개의 이써네트기판이 필요하다. 하나는 비보안망에 접속하는것으로 리용하고 다른 하나는 보안망에 접속하는것으로 리용한다.

조작환경

Linux 경로기쏘프트웨어를 시작하기 전에 경로기가 구성을 인식하고 수행할수 있도록 기동환경을 정의한다.

먼저IP주소를 결정하자. 비보안망은 망마스크 255.255.255.0와 함께 주소 200.200.200.0에서 200.200.200.255까지 리용한다. 외부세계의 망과 접속한 경로기는 200.200.200.1에 있고 만들려고 하는 경로기는 비보안측에서 보면 200.200.200.254를 가진다. 이것들은 인터네트의 거처지에 있는 실질적인 IP주소이다. 이것들은 인터네트에 존재하며 오직 하나뿐이다. 만일 외부세계에서 경로기접속을 허가한다면 인터네트에 있

는 콤퓨터는 보안하지 않은 LAN에 있는 임의의 주콤퓨터에로 직접 통신할수 있다.

보안망은 망마스크주소 255.255.255.0와 함께 IP주소 10.10.10.0부터 10.10.10.255까지를 리용한다. 보안 및 비보안 LAN을 접속하고있는 경로기는 보안측의 IP주소 10.10.10.1을 리용한다. 이것들은 비공개이거나 혹은 경로선택을 하지 않는 IP주소이다. 세계는 내부망에 있는 주콤퓨터들의 존재에 대하여 모른다. 그러므로 내부망에 있는 주콤퓨터를 가지고 직접통신은 불가능하다. 방화벽의 역할로부터 경로기는 보안망(비보안망을 포함하여)의 존재를 외부세계에 대하여 감춘다.

만들려는 경로기에서 장치 eth0은 비보안 LAN(IP주소 200.200.200.254)에 접속하며 장치 eth1은 보안LAN(주소10.10.10.1) 에 접속한다.

보안 LAN으로부터 수신접속이 없이 송신접속만 한다. 게다가 Linux핵심부의 망마스크화 (*IP masquerading*) 의 보안특성을 리용하여 망주소전송 (*network address translation*) 특성을 실행한다. 망주소전송에 대하여 경로기는 보안 LAN의 안과 밖으로 경로선택자료작업을 조종하지 못한다. 그러나 보안 LAN의 내부장치가 경로기를 통하여 접속되게 되면 역시 그 장치의 주소를 숨기고 비보안 LAN에 있는 주소안으로 그것을 전송한다. 이런 방법으로 세계에 대하여 보안망의 주콤퓨터들의 존재를 숨기면서 접속이 주콤퓨터의 주소에로 되돌이할수 없게 한다.

보안망이 경로선택이 없는 IP주소를 리용하기때문에 비보안망의 주콤퓨터들은 그 망의 주콤퓨터들의 존재에 대하여 모르며 서로 통신을 할수 없게 된다. 이와 비슷하게 인터네트에 비보안망을 접속한 경로기는 내부의 존재에 대하여 모르며 보안망과 경로기는 내부망의 주콤퓨터들에 직접 파케트들의 경로를 조종할수 없게 된다.

Linux 경로기시동

Linux경로기로 콤퓨터를 초기시동하자면 간단하게 이미 창조한 플로피초기시동디스크를 넣고 콤퓨터를 재시동한다. 콤퓨터는 플로피디스크로 시동한다.

그때 Linux경로기대상과제쏘프트웨어는 RAM디스크를 만들고 RAM디스크안에 RAM구동기를 창조하며 초기시동디스크의 내용을 보관한다. 그다음 경로기는 RAM으로부터 기동한다. 구성정보를 변화시켜 보존하고 있는 구성정보를 다음에 초기시동할수 있도록 초기시동디스크에 보관한다.

초기시동한 후 가입재촉문이 현시된다. Linux경로쏘프트웨어는 암호가 없이 뿌리사용자로 가입하고 그림 31-5에서 보여 준 구성차림표로 시동한다. 이미 서술된바와 같이 소규모적인 망을 기동하기 위한 단계들을 보기로 하자.

1. 망기판모듈들: 창조한 두번째 플로피디스크를 취급하자. /mnt/floppy등록부에 있는 플로피의 설치를 전제로 하면서 /mnt/floppy등록부에서 /lib/modules등록부까지 설치된 두개의 망기판들에 필요한 모듈을 복사한다.
2. 기초설정들: IP주소, 망마스크화, 두개의 이써네트기판들의 경로선택을 구성한다.
3. 기초보안: 망주소전송을 완성하는데 필요한 망보안을 구성한다.
4. 구체적구성: 이름봉사기(DNS) 와 주콤퓨터일람표와 같이 추가적인 망파라메터들을 구성한다.

이 차림표는 완전히 수동적으로 조종되는데 수자입력으로 선택하고 Enter건을 누른다. 실례로 주요구성차림표에서 Network Setting에 해당한 1을 입력하면 망설정보조차림표가 나타난다. 보조차림표의 입구점을 선택하면 간단한 편집기에서 (구성목적을 위하여) 편집파일을 연다. 편집기를 효과적으로 사용하기 위해 어떻게 보관하고(Ctrl+W 를 누르고 Enter건을 누르면 파일이름이 일치된다.) 탈퇴하는가(Ctrl+C를 누른다.)를 반드시 알아야 한다. 만일 이 지령들을 기억하지 않으면 편집하는 동안에 F1을 누르고 지령목록을 보아야 한다.

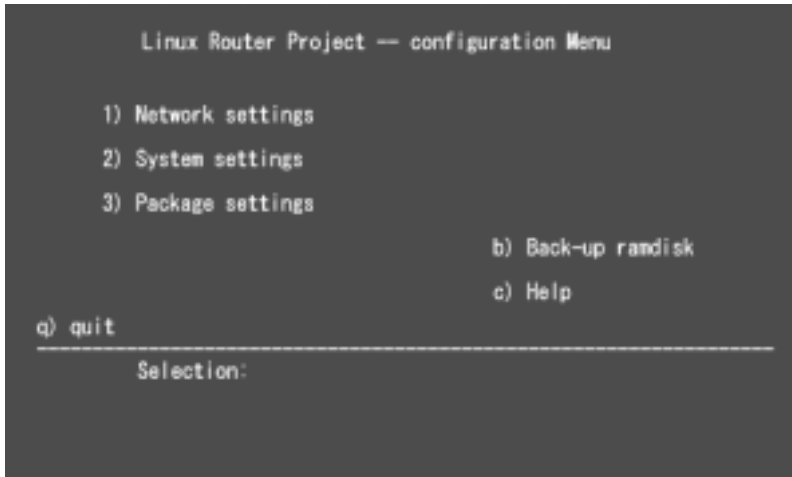



그림 31-5. Linux경로기대상과제 주요차림표

망기판모듈의 설치

Linux경로기대상과제초기시동디스크(창조한 첫번째 디스크)는 임의의 망기판모듈을 포함하지 않는다. 그 모듈들은 창조한 두번째 플로피디스크부분에 있다. 다음과 같은 단계들에서 Linux경로기대상과제초기시동디스크로 시동된 콤퓨터의 두번째 플로피디스크에서부터 RAM디스크까지에서 필요한 모듈들을 복사한다.

Linux경로기대상과제구성차림표로부터 다음과 같은 단계들을 처리한다.

1. Linux경로기대상과제차림표에서 탈퇴하려면 Q를 누른다.
2. mount /dev/fd0 /mnt지령으로 두번째 플로피디스크(망기판모듈을 가진)를 삽입하고 설치한다.
3. RAM디스크의 /lib/modules등록부에 필요한 모듈들을 복사한다. 실례로 ne2000 과 pcnet32구동프로그람을 요구하는 기판들이 있다면 cp /mnt/ne.o /lib/modules과 cp /mnt/pcnet32.o /lib/modules지령을 실행한다.
4. /usr/sbin/lrcfg지령을 리용하여 경로기대상과제구성차림표로 되돌이 한다.
5. /etc/modules파일을 구성하여 망기판구동프로그람을 추가한다. 이 파일을 호출하려면 첫 차림표에서 3을 선택하고, 두번째 차림표에서 2를 선택하며, 세번째 차림표에서 1을 선택한다. irq=10 와 io=0x300 와 같은 포구와 주소를 포함하는

pcnet32 와 ne와 같은 모듈들의 이름을 추가한다. 기정편집기에서 사용하는 지령의 목록을 보려면 F1을 누른다.

6. 필요한 모듈에 대한 망마스크 된 IP주소를 설정한다. 실례로 경로기를 통하여 FTP호출을 설정하려면 ip_masq_ftp의 앞에서 #을 지운다.
7. Alt+S지령으로 파일을 쓴다. Alt+Q지령으로 이 편집기에서 탈퇴한다. 두개의 Q지령을 리용하여 주요차림표로 돌아 갈수 있다.
8. 끝으로 구성을 검사하기 위하여 체계를 재기동한다. 플로피구동기에 Linux 경로기대상과제초기시동디스크를 넣는다. 콤퓨터는 설치된 망기판들을 탐색한다.

기초망설정구성

다음은 기초망설정을 구성하자. 경로기대면부는 그 망에 접속하였다. 사용자가 누구인가를 경로기에 알려 준다. 사용자대면부가 어떤 망기판에 접속하는가를 경로기에 알려 준다.

이를 위하여 기본망설정파일을 편집하여야 한다. 망의 선택항목 1(주요구성차림표의 선택항목 1에 의하여 호출된)은 보조차림표를 설정한다. 주요망설정파일은 편집기에서 연다.

설정한 경로기는 망에서 망으로 통보들을 전송한다. 따라서 전송한 IP를 설정하는것이 필요하게 되는데 이것은 다음지령행들을 NO로부터 YES로 변화시킬 필요가 있다는것을 의미한다.

```
IPFWDING_KERNEL=YES
IPFWDING_FW=YES
```

다음의 지령행을 NO로부터 YES 로 변화시키는 방법으로 경로기에 주콤퓨터이름을 할당할수 있는가를 확인한다.

```
CONFIG_HOSTNAME=YES
```

이제 IP주소를 할당하고 이써네트기판을 망마스크화한다.

지령행들을 찾아 보자.

```
#IFO_IFNAME=eth0
IFO_IPADDR=dhcp
IPO_NETMASK=255.255.255.255
IFO_BROADCAST=255.255.255.255
IFO_IP_SPOOF=YES
```

eth0를 정확히 설정하는 지령들을 변화시키고 첫 행에서 #를 지우는 방법으로 대면부를 능동적으로 만들어 본다.

```
IFO_IFNAME=eth0
IFO_IPADDR=200.200.200.254
IPO_NETMASK=255.255.255.0
IFO_BROADCAST=200.200.200.255
IFO_IP_SPOOF=YES
```

행들을 찾아보자.

#IF1_IFNAME=eth1
IF1_IPADDR=192.168.0.20
IF1_NETMASK=255.255.255.0
IF1_BROADCAST=192.168.0.255
IF1_IP_SPOOF=YES

마찬가지로 eth1(본질에 있어서 보안망)를 정확히 설정하는 지령들을 변화시키고 #를 제거한다.

IF1_IFNAME=eth1
IF1_IPADDR=10.10.10.1
IF1_NETMASK=255.255.255.0
IF1_BROADCAST=10.10.10.255
IF1_IP_SPOOF=YES

이러한 단계들은 경로기에 해당 이써네트기판들을 구성하여 할당된 망으로 통신하게 한다(이써네트기판들이 망에 정확히 물리적으로 접속되는가를 확인한다.).

다음의 문문서고는 주콤퓨터정보를 설명한다. 모든 주콤퓨터와 관련된 지령행들의 앞에 #를 추가한다. 이 콤퓨터가 오직 경로기로만 봉사하고 있다면 주콤퓨터정보를 할당하는것은 일반적으로 필요 없다.

#HOSTO_IPADDR=192.168.7.123
#HOSTO_GATEWAY_IF=default
#HOSTO_GATEWAY_IP=192.168.1.200
#HOSTO_IPMASQ=NO
#HOSTO_ IPMASQ_IF=default

내부와 외부망들에 대하여 망주소들이 완전히 할당되였는가를 확인한다. 기정파일은 하나의 망에 대하여 다음과 같은 지령들을 포함한다.

# NETO_NETADDR=192.168.0.0
NETO_NETMASK=$IF1_NETMASK
NETO_GATEWAY_IF=$IF1_IFNAME
NETO_GATEWAY_IP=default
NETO_IPMASQ=YES
NETO_IPMASQ_IF=$IFO_IFNAME

이것을 변화시켜 보안 및 비보안망의 구성을 반영하는 지령을 확인한다.

NETO_NETADDR=200.200.200.0
NETO_NETMASK=$IFO_NETMASK
NETO_GATEWAY_IF=$IFO_IFNAME
NETO_GATEWAY_IP=default
NETO_IPMASQ=NO # No need to masquerade real IP addresses
NETO_IPMASQ_IF=$IFO_IFNAME

```
NET1_NETADDR=10.10.10.0
NET1_NETMASK=$IF1_NETMASK
NET1_GATEWAY_IF=$IF1_IFNAME
NET1_GATEWAY_IP=default
NET1_IPMASQ=YES         # You must masquerade private IP addresses
NET1_IPMASQ_IF=$IF1_IFNAME
```

끝으로, 망에서 인터네트까지 기정경로기인 관문을 구성한다.

```
GW0_IPADDR=200.200.200.1
GW0_IFNAME=$IF0_IFNAME
GW0_METRIC=1
```

다른 부분의 구성파일은 추가적인 구성을 요구함이 없이 기정경로선택을 설정한다. 경로기에 주콤퓨터이름을 할당한다. 선택한 주콤퓨터이름을 경로기에 주려면 다음의 지령으로 변화시킨다.

```
HOSTNAME=tourettes
```

이제 이 파일을 보관한다. 보관은 Ctrl+W와 Enter를 눌러 하고 망구성차림표로 되돌이하려면Ctrl+C 을 누른다. /etc/network_direct.conf파일을 입력하려면 이 차림표에서 2를 누른다. 이 파일에서 기초보안을 설정할수 있다.

기초보안설정

다음 지령을 /etc/network_direct.conf파일에 추가한다.

```
[ "IF$" ] && ipfwadm -F -a -m -s "$NETWORK1"/24 -D 0.0.0.0/0
```

ipfwadm 문법(이것은 http://www.dreamwvr.com/ipfwadm/ipwadam-faq.html에서 찾을수 있다.)에로 들어 감이 없이 이 지령이 다음과 같이 한다는 것을 주목한다.

- 보안망에서 외부세계로의 개방적인 접속을 위하여 IP전송을 가능하게 한다.
- 보안망의 초기주콤퓨터주소보다는 차라리 모든 외부접속의 원천주소가 경로기의 외부IP주소로 되도록 마스크화한 IP를 허가하는것이 더 낫다. 외부에서 내부보안망의 주소를 볼수 없는 주소는 체계를 파괴시키기때문에 사용할수 없다.
- 모든 되돌이접속들을 조종하여 개방성접속에서의 응답자료가 보안망내부의 주콤퓨터로 정확히 되돌이하는가를 확인한다.

이제 파일을 보관(Ctrl+W)하고 편집프로그람에서 탈퇴(Ctrl+C)한다.

주의 가장 최신적인 ipchains에 대한 기초보안기술은 이 책의 범위를 넘어 문서고 있다. 더 많은 정보를 보려면 lrp.c0wz.com 인 Linux경로기대상과제문서Web싸이트안에서 보안련결을 참고하도록 한다. 이 스크립트에 대한 하나의 좋은 실례는

seawall.sourceforge.net에서 리용할수 있는 "Seattle Firewall" 이다.

더 먼 곳에서 망구성

더 먼 곳에서의 망구성은 필요 없다. 실례로 hosts.allow 와 hosts.deny 파일을 보자. 이 파일들은 망을 통하여 간접적으로 경로기에 가입하려는 부정적시도의 공격으로부터 경로기를 보호하기 위해 설계하였다. 다시말하여 경로기는 망을 통해서 가입하기보다 조종탁에서 구성하는것이 더 필요하다. 아마 변하지 않은 파일들은 남기는것이 좋다.

변화된 정보의 보관

일단 경로기를 구성하면 초기시동플로피디스크에 변경된 내용들을 보관할 필요가 있다. 이것은 주요구성차림표에서 b를 선택하는 방법으로 할수 있다. /etc등록부에서만 파일을 변화시켰기때문에 etc모듈을 보관하고 있다. 이것은 보관차림표에서 2를 선택하는 방법으로 실현할수 있다.

변경을 초기시동플로피디스크에 쓴후 새 구성이 가능한 체계를 재시동한다. 이로부터 기초방화벽으로 봉사하는 경로기기능을 가지게 된다.

더 멀리 가기

이 기초원리들을 리용하면서 Linux경로기대상과제배포물로 작업하여 고속도인터네트접속과 ISDN접속, 전화접속방식 PPP사용자를 지원하는 더 선진적인 (다른것들중에서)경로선택을 진행한다. 이 모든것들은 모듈의 추가적인 결합과 구체적인 체계구성을 요구한다. 이에 대한 더 많은 정보는 http://www. linuxrouter.org에서 Linux경로기대상과제Web싸이트에서 얻을수 있다.

▶미리 알아두기

이 장에서는 유연한 Linux가 위력한 워크스테이션이나 망봉사기로가 아니라 경로기와 보안방화벽으로 어떻게 PC를 구성하는가를 보았다.

제32장 《Web봉사기구성》에서는 인트라네트 Web봉사기로 Linux를 어떻게 사용하는가 하는 상세한 론의와 함께 망환경안에서 Linux에 대한 고찰을 계속한다. Linux는 저가격콤퓨터에서 기동할수 있는 고성능, 표준공업Web봉사쏘프트웨어의 리용성으로 하여 인터네트와 인트라네트Web봉사기들에서 더욱더 인기가 높다.

사실 LinuxWeb봉사기들의 믿음성과 속도는 높은 비용의 봉사기와 상업적Web봉사기들이 리용할수 있는 많은 세부항목들에서 인기가 있다.

만일 Web 개발이나 설계에 대한 예비지식이 없다면 그 장에서는 자체의 Web 봉사기설정을 시작하는데 필요한 기초지식을 계속 제공해 준다.

제 7 편. Red Hat linux 7.1 을 Web 와 전자우편봉사기로 리용

제 32 장. Web 봉사기구축

지금까지 Linux 에 대한 학습을 많이 하였으므로 이제는 Linux 의 가장 인기 있는 응용의 한가지 즉 소형 및 중형 Web 봉사기의 가동환경으로서 조작체계를 리용하는 방법에 대해 고찰할수 있을것이다.

사실 Linux 는 여러가지 리유들로 하여 Web 봉사기용으로 널리 리용되고 있다.

- Unix 의 유연성과 관리에서의 편리성을 보장한다.
- 설치하고 실행하는데 비용이 거의 들지 않는다.
- 아주 정교한 Web 싸이트들을 만들수 있게 하는 수많은 도구들을 제공한다.

이 장에서는 Linux 가 설치된 콤퓨터를 인트라네트와 인터네트 싸이트용 Web 봉사기로 전환하기 위한 기초지식들을 서술한다. 여기서는 Web 봉사기에 의하여 처리되는 일감전체를 개괄하는것으로부터 시작하여 Linux 에서 흔히 쓰이는 주요 Web 봉사기들의 일부를 시험한다.

뿐만아니라 인터네트우에서 실지로 가장 대중적인 봉사기로 되고 있으며 Red Hat Linux 7.1 에 현실적으로 태워 져 있는것들중의 하나인 Apache Web 봉사기를 설치하고 구성하며 관리하는데 필요한 구체적인 정보들을 취급한다.

이 장은 Apache 를 리용해서 간단히 Web 싸이트를 만드는 과정을 보여 주는것을 통하여 전체 과정을 안내하는것으로써 끝마친다. 이 장의 마감에 이르면 Linux 와 Apache 를 리용하여 자체의 Web 싸이트를 만들수 있는 수준에 올라야 하며 Linux 에서 쓸수 있는 다른 Web 봉사기들을 자신 있게 써볼수 있어야 한다.

Web 봉사기란 무엇인가

Web 를 사용해 본적이 있다면 아마도 《Web 봉사기》라는 용어를 많이 들었을수 있다. 그러나 Web 봉사기가 무엇을 하는가에 대한 명확한 표상은 가지고 있지 못할것이다.

한가지 용어가 흔히 두가지 의미 즉 물리적인 장치와 그것들을 움직이는 쏘프트웨어라는 개념으로 쓰이는 경우가 있지만 《Web 봉사기》는 정확히 Web 열람기들과 같이 Web 의뢰기로부터 들어 오는 질문들에 대답을 주도록 실행되는 쏘프트웨어를 말한다.

Web 봉사기쏘프트웨어는 단순한 목적 즉 그 쏘프트웨어에 의하여 조종되고 있는 Web 싸이트로부터 얻을수 있는 문서들에 대하여 하이퍼문서이송규약(HTTP)에 따라 의뢰기들의 질문들에 대답을 주도록 설계된다.

이것은 다음과 같이 진행된다. 의뢰기는 URL 을 리용하여 문서를 요구한다. 그러면 Web 봉사기는 그 요구를 접수하고 체계우의 물리적파일(그것은 HTML 파일이거나 기타 다른 류형의 파일중의 하나일수 있다.)에 URL 을 대응시킨다. 그다음 의뢰기가 그 파일을 회수할수 있는 권한을 가지고 있는가를 확인한후에 그 파일을 의뢰기에 돌려 보낸다. 게다가 Web 봉사기는 일반적으로 누가 어떤 문서들을 요구하였고 몇번 요구들이 들어 왔는가를 등록하고 통계한 결과에 따라 Web 싸이트들이 얼마나 일반적인

가를 결정하는데도 리용하게 된다.

파일의 고속회복

우의 설명은 전체적으로 놓고 보면 간단하지만 그것은 대다수의 Web 봉사기들이 하는 일들을 설명하였다.

물론 Web 을 열람할 때 Web 봉사기가 Web 열람기들의 요구에 따라 거기에 보내는 정적문서들의 모임에 불과하다는것을 인차 느끼게 된다. 양식들은 봉사기로부터 정보를 요구하거나 봉사기를 운영하고 있는 조직에 정보를 제공하는데 리용될수 있다. 상품을 주문할수 있고 신용기판을 검사할수 있으며 여러가지 다른 류형의 업무처리를 수행할수 있다.

이 모든 상호작용성이 보장되도록 하기 위하여서는 현대 Web 봉사기들이 단순히 HTTP 요구들에 대답을 주는데 그쳐서는 안된다. Web 봉사기들은 일반적으로 호상작용을 위하여 두가지 수행기구를 제공한다.

- 공통관문대면부(CGI)
- 봉사기응용프로그람대면부(API)

공통관문대면부

CGI 는 Web 봉사기에 상호작용성을 추가하는데 가장 널리 사용되는 방법이다.CGI 모형하에서는 정적파일들을 요구하는 HTTP 가 아주 간단히 확장된다.

CGI 는 프로그람이 봉사기우에서 실행되게 하며 양식의 자료가 처리용프로그람에 통과되도록 하는 표준화된 방법들을 제공해 준다. 이러한 프로그람들은 임의의 프로그람작성언어 또는 스크립트언어로 작성할수 있다. 보통 C, Perl, Java 를 리용한다.

사용자가 CGI 프로그람(양식을 제공하거나 프로그람에 련결시킬수 있는)을 요구할 때 Web 봉사기는 사용자의 자료를 CGI 프로그람에 통과시키고 그 프로그람이 자료를 복귀시키기를 기다린다. 프로그람에 의하여 복귀되는 자료는 정적파일의 내용이 열람기에 복귀되는것과 같은 방법으로 의뢰기로 직접 되돌아 간다. 유효한 내용이 의뢰기에 복귀되도록 열람기에 복귀되는 유효한 내용들을 작성하고 모든 우발적인 사건들을 조종하는것이 그 프로그람의 일감이다.

CGI 개념은 전반적으로 잘 작용한다. 자료가 봉사기로부터 CGI 프로그람에로 넘어가는 방식과 프로그람이 봉사기에 복귀시키는 자료를 만드는데 필요한 방식의 단순성은 간단한 CGI 프로그람을 약간한 프로그람작성경험을 가지고도 작성할수 있다는것을 시사해 준다.

더우기 CGI 프로그람을 변경하고 시험하는것이 상당히 쉽다. 그것은 Perl 과 같은 인기 있는 스크립트언어를 써서 그것을 작성할수 있기때문이다.

CGI 대면부의 표준속성은 또한 만일 그것이 모든 조작체계에 있는 일반적인 언어를 리용하여 작성되였다면 하나의 Linux Web 봉사기를 위하여 작성된 CGI 스크립트 혹은 프로그람이 다른 Linux Web 봉사기 그리고 가능한 다른 Unix 봉사기우에서 변경함이 없이 자기의 기능을 수행할수 있을것이라는것도 시사해 준다.

이러한 많은 우점들과 반면에 CGI 는 일부 Web 싸이트들에서는 대응하지 못하는 일부 심각한 약점을 가지고 있다. CGI 의 두가지 기본결함은 안전성과 속도에 관련된다.

Web 가 출현한 때로부터 만일 스크립트를 정교하게 작성하지 못하면 Web 봉사기를 실행하고 있는 체계가 약삭 빠른 해커에게 완전히 호출될수 있는 CGI 대면부에는 안전상

많은 결함들이 나타나게 된다. 이것은 거의 모든 단체의 인터네트와 인트라네트 봉사기 특히는 금융거래와 신용기판판매에 대한 직결봉사를 제공해 주는 싸이트에서처럼 Web 봉사기상의 자료안전을 보장하여야 하는 체계에서 CGI 가 기대하는것만 못하다.

더우기 CGI 대면부는 그리 효과적인것이 못된다. Web 봉사기는 의뢰기의 요구들에 대답을 주는 하나이상의 프로쎄스들을 실행한다. 열람기는 그다음 CGI 프로그람용자식프로쎄스들을 시작하고 자료들을 이 새로운 프로쎄스에 넘겨 주고 그것이 끝나기를 기다린다. 복잡한 싸이트에서는 이것이 짧은 시간내에 시작해야 할 대단히 많은 새로운 프로쎄스들을 생성하게 하는데 특히 CGI 스크립트를 대대적으로 리용하고 있는데서는 더더욱 그러하다. CGI 프로그람에 대한 매 요구는 그에 해당한 별도의 프로쎄스를 만들어 낸다.

이것은 거대한 량의 자료와 요구들을 처리하기에는 상당히 불합리한 방법이며 대부분의 인기 있는 Web 봉사기들이 봉사기측 프로그람작성에 자체의 API 들을 실현하고 있는 리유로 된다.

응용프로그람대면부

API 들은 Web 봉사기안에 긴밀히 통합되는 프로그람들을 작성하는 방법을 제공해 주며 일반적으로 요구마다 새로운 프로쎄스를 필요로 하지는 않는다. API 들은 류사한 CGI 프로그람들과 비교될만큼 많은 요구들을 조종할수 있는 능력을 가진 Web 응응용프로그람들을 개발할수 있게 한다. 더우기 API 로 작성한 프로그람들은 안전성측면에서 비평이 적은 대상으로 되여 왔다.

API 에 기초한 프로그람들은 일반적으로 CGI 프로그람이 하는것과 같은 일감들을 수행할수 있다. 즉 양식들안에 제공된 정보들을 처리하고 자료기지의 자료를 호출하며 신용카드들을 검사하는것 등의 일감들을 처리할수 있다.

봉사기 API 들의 가장 큰 결함은 그것들이 응용프로그람을 특정의 Web 봉사기에 의존하게 한다는것이다. 어떤 응용프로그람을 새로운 봉사기에 옮기는것은 상당한 정도로 프로그람을 다시 작성할것을 요구할수 있는데 이것은 큰 규모의 응용프로그람에서 실용적이지 못한것으로 될수 있다.

봉사기의 내장 API 를 리용하여 봉사기우에 큰 응용프로그람을 전개하기전에 안전성, 성능,우연성에 관하여 봉사기를 잘 검사하여야 한다는것을 잊지 말아야 한다.

Linux Web 봉사기

인터네트의 골격이 많은 Unix 를 개발한 사람들에 의하여 발생되였기때문에 Unix 와 Linux 가동환경들이 넓은 범위의 각종 Web 봉사기로서 리용되고 있다는것은 놀라운것이 아니다.

Unix 세계안에 대다수의 것들과 마찬가지로 Web 봉사기분야에서도 Linux 가 리용가능하다. Linux Web 봉사기들의 대부분은 무료이다. 가장 널리 알려 진 Linux Web 봉사기들은 다음과 같다.

- NCSA httpd
- Apache
- AOLserver
- Boa
- WN
- W3C/Cern

이러한 무료봉사기들외에 Linux 에서 리용가능한 여러가지 변종의 판매제품들이 있는데 다음과 같다.

- FastTrack/iPlanet
- Java Web Server
- Stronghold
- Zeus

Apache

일부 통계에 의하면 Apache 는 가장 널리 쓰이고 있는 Web 봉사기쏘프트웨어이다. Apache 는 《A Patchy Server》의 략어이며 초기의 Web 봉사기들중의 하나인 NCSA httpd 의 일부 문제점들을 해결하고 일부 기능들은 추가하도록 개량하려는 노력에 의해 발전하였다.

그때부터 Apache 는 Unix 체계를 위한 비상업적인 봉사기로서 출현하였다. 보다 최근에는 Windows 에로 이식되여 Windows NT/2000 체계의 Web 봉사기로써도 리용될수 있다.

Apache 는 Unix 및 Linux 체계관리자들의 흥미를 끄는 여러가지 특징을 가지고 있다. 초기의 NCSA httpd 구성파일들에 기초한 구성을 리용한다는것외에 Apache 는 많은 다른 인기 있는 Linux 용 응용프로그람들과 같이 완전한 원천코드도 제공되며 집체적으로 개발할수 있다.

Apache 는 자기자체의 API 를 제공하는데 그것은 CGI(그것은 역시 Apache 에 의하여 지원된다)대신에 리용될수 있다. 뿐만아니라 API 는 여러가지 목적에 봉사하는 plug-in 모듈들을 만드는데도 리용될수 있다. 다음의 모듈들이 리용가능하다.

- NIS 인증봉사기들 혹은 라이트급등록부호출규약(LDAP:Lightweight Directory Access Protocol)자료기지로부터의 인증을 비롯한 선택적인 인증체계들
- PHP/FI 와 HeiTML 을 비롯한 Microsoft Active Server Pages 혹은 Netscape LiveWire 와 같은 기능을 봉사하는 봉사기측서술환경
- 전통적인 CGI 의 성능을 개선하기 위하여 설계된 모듈들. 례를 들어 FastCGI 모듈은 CGI 프로그람을 실행시키고 결과들을 되돌려주는데 걸리는 시간을 최소화하기 위한 간단한 방법들을 리용한다. Perl 모듈은 Perl 스크립트들이 단일프로쎄스안에서 실행되고 처음 실행에서만 콤파일되게 하는데 이것은 Perl 에 기초한 CGI 가 콤파일된 CGI 프로그람들이나 일부 API 에 기초한 Web 응용프로그람들과 거의나 같은 속도로 실행되도록 한다.

Apache 의 마지막 판본에 대한 원래의 원천코드는 httpd.apache.org 로부터 리용할수 있는데 Linux 를 포함하는 많은 여러가지 체계들을 위하여 2 진코드로 미리 콤파일해놓은것과 함께 있다. Apache 는 Red Hat Linux 7.1 과 함께 설치되는 기정 Web 봉사기에 포함된다.

NCSA httpd

NCSA httpd 는 Web 가 처음으로 만들어 진 초기의 두개의 Web 봉사기들중의 하나(다른 하나는 Cern Web 봉사기이다.)이다. NCSA httpd 는 Web 가 광범한 인기를 얻을수 있게 떠밀어 준 초기의 도형방식 Web 열람기인 Mosaic 의 고향이기도 한 Urbana-Champaign 에 있는 Illinois 종합대학에서 특급콤퓨터응용프로그람을 위한 국립

썬터에 제출되였다.

NCSA httpd는 Web싸이트들의 거의 모든 요구를 만족시키도록 설계된 핵심함수모임을 제공한다. 이것들의 특징은 다중주콤퓨터에 대한 내정지원, Basic와 Digest 인증, 등록부준위접근조종, 봉사기측 포함, 완전한 CGI지원이다.

NCSA httpd가 더이상 활발히 개발되지는 않지만 그것은 Apache의 현구조에 대한 배경으로 된다. NCSA httpd 의 초기원천코드와 그것을 콤파일한 2 진코드들은 hoohoo.ncsa.uiuc.edu로부터 리용할수 있다.

W3C/Jigsaw

Jigsaw Web 봉사기는 인터네트의 동력으로 되였으며 첫 Web 봉사기들중의 하나였던 Cern의 Java로 작성한 계승판이다. 완전한 Java위주의 봉사기로써 그것은 Unix 및 Linux와 Microsoft Windows를 포함하여 이 언어를 지원하는 그 어떤 조작체계우에서도 실행되도록 설계되였다.

Jigsaw는 다음과 같은 여러가지 형태의 Java객체들에 의하여 설치된다.

자원 Web 페지에서 보이는 것. 이것은 본문이나 화상파일과 같은 정적객체들과 스크립트들과 같은 동적객체들을 포함한다.

프레임 실지 자원을 조종하는 것. 프레임은 특수한 자원을 봉사(례를 들어 HTTP 프레임객체는 HTTP자원을 봉사한다) 하는데 필요한 모든 정보들을 포함한다.

려파기 자원을 동적으로 수정하는 수단. 실례를 들어 만일 Web 싸이트가 사용자가 가입되였다는것을 보지 못한다면 가입등록페지에서 《려파》시킬수 있다.

색인작성기 자원분류를 위한 수단. 두개의 기본색인들은 등록부(파일을 묶어 놓기 위한것)와 확장자(TXT나 INI와 같은 일반 파일용)이다.

이 책의 편집 당시 리용할수 있는 안정한 판본은 Jigsaw 2.2.0이다. Jigsaw 판본수자들은 Linux 핵심부판본들과 류사한데 실례로 앞으로 Jigsaw 2.3.4와 같이 가운데 수자가 홀수인 판본은 안정하지 못하다. W3C 는 www.w3.org/Jigsaw에서 봉사기에 대한 페지를 관리한다.

주의	앞에서 말한 W3C 봉사기, Cern은 인터네트우의 초기 eb봉사기들중의 하나였다. 유감스럽게도 더이상 지원되지 않고 있다.

WN

무료봉사기들에 대한 이 장의 개괄은 그다지 널리 리용되지 않는 봉사기들쪽으로 옮겨 지고 있다. 그러나 이러한 지위의 시장봉사기들은 각이한 Web봉사기기술과 특징들을 볼수 있게 한다.

WN은 다른 Web봉사기들과는 달리 독특한 특징을 가진 또 하나의 무료봉사기이다. 례를 들어 WN 은 개발자가 다중련결파일들로 구성된 문서인 론리적인 HTML 문서라는 것에 대한 full-text 탐색을 가능하게 한다. 게다가 사용자들은 봉사기우에서 파일들을 탐색할수 있으며 쉽게 일치되는 문서들을 얻을수 있다. 사용자들은 또한 다중련결된 문서들로 만들어 진 단일한 론리적문서를 내리 적재할수도 있는데 그것은 사용자들이 작은 문서계렬들로 구조화된 파일들을 인쇄하기 쉽게 만든다.

WN 봉사기의 다른 하나의 독특한 특징은 조건부문서들을 봉사하는 능력이다. 즉 의뢰기의 IP 주소나 의뢰기의 열람기판본과 같은 가변적인 요소에 기초하여 정확한 판본이 의뢰기에 보내지도록 하는 정의에 맞추어 단일문서를 창조할수 있다.

상대적으로 작은 규모는 물론 WN 봉사기의 안전모형 역시 그것이 Apache 나 NCSA httpd Web 봉사기계렬과 차이나는 점이다. Apache 나 NCSA httpd Web 봉사기들에서는 권한이 특별히 부정되지 않는 한 파일을 봉사한다. 그러나 WN 에서는 권한이 파일에 대해 부여되지 않는한 아무런 파일도 봉사하지 않는다. 이것은 봉사기를 보다 더 안전하게 만들며 파일호출에 대한 보다 세밀한 조종을 할수 있게 한다. hopf.math.nwu.edu 에서 WN 의 복사품을 얻을수 있다.

Boa

봉사기에 대한 개괄에서 마지막 무료봉사기는 Boa 봉사기이다. 이것은 이 책에서 서술하기 이전에 한번정도밖에 나오지 않은 거의 알려 지지 않은 봉사기이다.

Boa 는 봉사기들이 작고 간단하며 기초적인것들로 될수 있으며 그래도 여전히 유익한 역할을 할수 있다는것을 보여 주기 위하여 여기에 포함시켰다. Boa 는 가장 기초적인 기능들과 Apache 나 WN 보다 덜 섬세한 접근조종을 제공한다.

그러나 Boa 는 Linux 용의 다른 거의 모든 Web 봉사기들보다 가능한한 보다 빠르게 동작하도록 하기 위하여 설계되였다. 사실 Boa 의 창조자들은 그 봉사기가 Apache 보다 휠씬 더 빠르다고 주장하는데 이 주장을 증명하기 위해서는 큰 Web 싸이트들에 대한 보다 실제적인 세계에서의 검사가 필요하다.

Boa 는 단일과제설계를 통하여 성능향상을 얻고 있다. 전통적인 Web 봉사기들은 요구들을 접수하고 매 요구를 조종하기 위한 프로쎄스들을 만드는데 Boa 봉사기는 조작체계의 다중과제기구가 수행되도록 하는것이 아니라 단일프로쎄스로써 기동하여 내부적으로 다양한 요구들의 모든 처리 및 실행을 조종한다. Boa 는 CGI 요구가 만들어 졌을 때만 자식프로쎄스를 산생시킨다. Boa 에 대해 더 구체적으로 알려면 www.boa.org를 보아야 한다.

Stronghold

Stronghold 는 Linux 에 있어서 가장 널리 알려 진 판매용 Web 봉사기들중의 하나이다. Stronghold 는 아주 안전하고 충분한 지원을 받는 쏘프트웨어제품을 제공하는 Apache 의 판매용판본이다.

Stronghold 는 권한증명도구들을 포함하여 안전한 봉사기를 설치하는데 필요한 모든 도구들을 제공한다. 필요하다면 권한증명서를 리용하여 VeriSign 같은 제 3 자의 권한증명서에 의하여 증명된 수자증명서를 발행할수 있다. Stronghold 의 또 다른 안전특징은 안전을 보장하기 위하여 128bit 자료암호화와 같은 장치적인 고속암호화에 대한 지원을 보충적으로 포함하는것이다.

더우기 Stronghold 가 완전한 원천코드를 가지고 있기때문에 Apache 모듈들에서의 콤파일하기와 자기자체의 모듈들을 작성하는것을 포함하여 Apache 를 리용하는 모든것에 대하여 그것을 리용할수 있다. Stronghold 는 현재 http://www.c2.net/products/sh3/index.php3에서 Red Hat 에 의하여 배포되고 있다.

FastTrack/iPlanet

FastTrack 는 Netscape 의 널리 알려 진 Web 봉사기계렬들중의 하나이다.

자체의 열람기와 열람기원천코드를 공개령역안에 공개하려는 자체의 최근 결정이 있을 때까지 Netscape 는 공교롭게도 Linux 와 같은 비상품적조작체계용 Web 봉사기들을 개발하는데 흥미를 가지고 있지 않았다.
AOL 및 Netscape 와 Sun 사이의 련합이 출현함으로써 FastTrack 를 회복하는데 관심을 가지게 되였다. FastTrack 는 현재 Web 와 응용봉사기들의 iPlanet 계렬의 한 부분이다. FastTrack 는 현재 iPlanet Web 봉사기의 Java 로 작성된 무료판본이다. 이 봉사기의 Enterprise 판본은 판매품이다.
FastTrack 는 관리하기 쉬운 Web 봉사기이다. 모든 구성과 관리는 Web 에 기초한 양식들을 통하여 집행된다. 이러한 양식들은 Web 주인이 사용자들을 만드는것으로부터 등록부준위에서의 파일들에 대한 접근을 조종하는것에 이르기까지 모든것을 다 할수 있게 한다.
FastTrack 는 봉사기측이 CGI 와 응용프로그람들을 개발하기 위한 Netscape 자체의 API 를 포함하도록 지원한다. 더우기 FastTrack 는 Netscape 의 봉사기측 Java 스크립트 개발환경인 LiveWire 를 지원할수 있다.
봉사기는 또한 SSL기술에 기초하여 안전한 봉사기로써 리용될수 있는 능력을 가지고 있다. FastTrack 에 대한 보다 구체적인 정보를 얻자면 http://home.netscape.com/fasttrack/v3.0/index.html에 가보아야 한다.

AOLserver

AOLserver 는 1995 년에 판매용으로 제공된 때로부터 현재 리용할수 있는 가장 완전한 특징을 가진 무료봉사기들중의 하나로 될 때까지 흘러 온 재미나는 력사를 가지고 있다.
AOLserver 는 초기의 유용한 판매용봉사기들중의 하나인 NaviPress 로 리용되기 시작하였다. 1995 년말에 AOL 은 NaviPress 를 만드는 회사를 가지게 되였으며 내부적으로 봉사기를 리용하기 시작하였다. 1997 년에 AOL 은 인터네트우에서 배포하기 위한 무료제품인 봉사기로 출현하였다.
현 시점에서 AOLserver 의 마지막제품(3.4)은 아래와 같은 특징을 가진다.

- Web 싸이트우에서 탐색기능을 보장하기 위한 내장 full-text 색인붙이기
- C 프로그람작성과 Tcl 스크립트언어로부터 호출할수 있는 내장 API
- 완전한 CGI(선택적이다)와 봉사기측지원포함
- Microsoft 의 Active Server Page(능동봉사기페지)들과 류사한 봉사기측동적페지기능
- 제 3 자의 Web 자료기지통합도구들이 없이 자료기지구동형동적응용프로그람을 구축하기 위한 자료기지결합성

현 시점에서 iPlanet 계렬봉사기들우에서 AOL 및 Netscape 와 Sun 사이에 련합을 형성하려는 노력에도 불구하고 AOLserver 는 여전히 지원되고 있다. www.aolserver.com에 있는 AOLserver Web 싸이트로부터 AOLserver 를 내리 적재할수 있다.

Java Web Server

Java Web Server 는 서술된 봉사기들중에서 Sun 의 JavaSoft 분과에서 만든 유일한 Web 봉사기이다. W3C/Jigsaw 봉사기와 같이 Java Web Server 는 완전히 Java 로서

개발되였고 리론적으로 Java 가상기계로서 임의의 가동환경우에서 실행할수 있다.

Java Web Server 는 CGI 와 SSL 을 위한 완전한 지원에 의거함이 없이 응용프로그람들을 작성할수 있게 하는 봉사기애플레트골격을 비롯하여 봉사기가 상점에서의 직결봉사와 같은 안전성을 요구하는 응용프로그람으로 리용될수 있도록 경쟁적인 업무용제공을 하게 하는 여러가지 특징들을 가지고 있다.

성능개선을 위하여 HTML 과 프로그람코드들을 둘 다 포함하는 콤파일된 페지들을 만드는것도 가능하다. 통계를 낼 목적으로 내장된 봉사기추적방식을 리용하여 사용자대화조종을 프로그람안에서 추적할수 있다.

AOL 및 Netscape 와 Sun 사이 련합의 장성으로 해서 유감스럽게도 그것을 지원하여야 할 세개의 Web 봉사기들이 생겨 났다. 현재 2001 년 초엽에 이 제품에 대한 개발을 중지시키기로 결정되여 있다. 그럼에도 불구하고 이것은 아직까지도 Web 봉사기을 위한 대중판매항목으로 되고 있다. 만일 Java Web 봉사기를 운영하거나 지원해야 한다면 http://www.sun.com/software/jwebserver /index.html 의 Java Web 봉사기 홈페지에서 그에 대한 상세한 설명서를 읽을수 있다.

Zeus

봉사기에 대한 개괄에서 마지막으로 론의되는 Web 봉사기는 Zeus Web 봉사기이다. Zeus 의 최신판본인 3.3.8 은 Unix 및 Linux 와 Macintosh OS X 를 위한 Web 봉사기이다. 그것은 내장된 클라스터화기능의 지원을 보장하는데 그것은 대용량싸이트용으로 Zeus 를 선택할수 있게 한다. 클라스터화에 의해 다중 Web 봉사기들로부터 단일 Web 주소를 봉사할수 있다. 이것은 요구의 적재를 봉사기그룹우에 분산시킬수 있게 하는데 그것은 보다 많은 동시에 발생하는 충돌을 조종할수 있게 한다는것을 의미한다.

이러한 기능은 Zeus 가 eBay 와 British Telecom 을 포함하여 일련의 가장 많이 사용되는 Web 싸이트들을 위해서 선택할수 있는 Web 봉사기로 되도록 하였다. Linux 2.4 핵심부와 관련된 일련의 발전은 Zeus Web 봉사기의 능력을 강화시킬것을 제안하고 있다. 최근기사에서 IBM Linux 기술쎈터는 Linux 2.4 핵심부에 기초한 봉사기들우에서 Zeus 의 성능은 Linux 2.2 핵심부에 기초한 봉사기들우에서 보다 거의 두배였다는것을 발견했다.

더우기 Zeus 는 Microsoft 의 인터네트정보봉사기로부터의 응용프로그람대면부인 ISAPI 에 대한 지원을 보장하며 Windows NT/2000 과 마찬가지로 Unix 우에서도 이 지원을 보장한다. Zeus 는 또한 Java Web 봉사기에서 리용된것들과 같은 Java servlet 들을 지원하며 자료기지접근을 위한 JDBC 표준을 리용하여 통합자료기지의 결합성을 보장한다.

아마 가장 흥미 있는것은 Zeus 가 세계적범위에서 완전한 128bit 암호화를 가진 SSL 을 제공한다는 점일것이다. 미국정부가 128bit 암호화기술에 대한 수출을 제한하기 때문에 미국회사들로부터 수출되고 있는 대다수 안전제품들은 안전성이 그리 높지 못한 56bit 암호화만을 제공하였다. Netscape 와 같은 제품의 암호를 해독하는 사람들의 출현에 대한 대다수의 보도는 56bit 암호화판본제품들과 관련되여 있다. Zeus 정보는 www.zeus.co.uk에서 리용할수 있다.

Apache 설치

이장의 나머지 부분에서는 다음의 두가지 리유로 해서 Apache Web 봉사기를 더 구체적으로 서술한다.

1. Apache 봉사기는 대다수의 Linux 배포물(이 책에 따라 나온 CD-ROM 우

의 Red Hat Linux 7.1을 포함하여)과 함께 실리여 오는 기정 Web 봉사기이다.

2. Apache가 현재 인터네트우에서 가장 인기 있는 Web 봉사기이기때문에 그것은 사용자들이 사용하고 싶어 하는 가장 적합한 Web 봉사기일것이다.

만일 Linux체계를 설치할 때 완전한 설치를 진행했다면 아마 이미 Apache를 설치하였을것이다. 이것을 검사할수 있는 몇가지 방법들이 있다. 실례로 rpm 지령을 리용할수 있다.

```
$ rpm -q apache
```

만일 Apache를 설치하였다면 다음과 같은 결과를 받을것이다.

```
apache-1.3.19-5
```

만일 Apache가 설치되였다는 통보를 받으면 《Apache 구성》이라는 절로 넘어 갈수 있다.

Red Hat CD-ROM에 의한 Apache의 설치

Apache를 설치하는 가장 단순한 방법은 이 책과 함께 따라 온 Red Hat CD-ROM을 가지고 설치하는것이다. 이를 위하여 간단히 CD-ROM을 설치(일반적으로 /mnt/cdrom 에)하고 그다음 cd /mnt/cdrom/RedHat/RPMS 지령을 리용한다.(만일 /mnt/cdrom과 다른 위치에 CD-ROM을 설치하였다면 설치점에 기초한 적당한 등록부로 변경시켜야 한다.) 이 등록부는 완전한 Red Hat Linux 배포물들을 위한 RPM 파일들을 모두 포함한다. 여기서 제품을 설치하기 위하여 간단히 rpm을 리용한다.

```
$ rpm -i apache-1.3.19-5.i386.rpm
```

최신판본 Apache의 적재

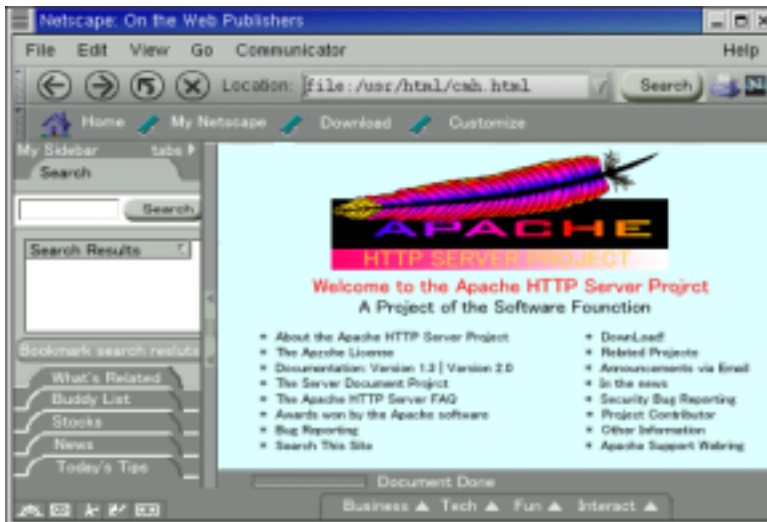



그림 32-1. Apache Web 싸이트

만일 중요한 사명을 맡은 또는 큰 용량의 Web 봉사기를 실행하려고 한다면 httpd.apache.org로부터 Apache의 최신판본을 내리적재하려고 할것이다.(그림 32-1을 참

고) 최신판본을 리용하면 보통 싸이트에 영향이 없는 작은 오유들이 아주 분주하고 중요하게 리용되는 싸이트에 영향을 주지 않는다것을 확인할수 있다.

주의

> Apache 의 구성과 관리에 대한 모든 론의는 이 책에 포함된 Red Hat CD-ROM 우의 Apache 판본을 리용한다는것을 전제로 한다. 일반적으로 Apache 의 새 판본들은 뒤방향호환성을 보장한다. 즉 우리가 이장에서 보게 되는 구성파일들은 Apache 의 보다 새로운 판본들과 함께 동작해야 한다.

Apache Web 싸이트로부터 Apache 를 적재할 때 원래의 원천코드를 적재하겠는가 아니면 Linux 용으로 이미 콤파일된 2 진파일들을 적재하겠는가 하는 항목이 있다. 원천코드판본을 리용하여야 할 때가 있지만 특수하게 만일 Apache API 를 리용하여 프로그람들을 개발하려고 하거나 자기 봉사기안에 외부모듈들을 구성하려고 한다면 기본적인 Apache 특징모임을 가지고 있는 Web 싸이트를 설치하기 위하여 이미 콤파일된 2 진파일들을 내리적재해야 한다. 이 책에서는 Apache 자체의 복사물을 콤파일하는것은 론의하지 않았는데 왜냐하면 이것이 한개 장에서 취급하기에는 Web 봉사기들에 대한 너무도 깊은 론의를 요구하기때문이다.

만일 Apache 로부터 최신 2 진배포물을 얻으려고 한다면 열람기가 Apache Web 싸이트(httpd.apache.org)를 가리키게 하고 거기서 Download 련결을 선택해야 한다. 이것은 파일들의 등록부에 들어 간다. 흔히 최종공개물은 이 등록부안에서 찾을수 있다. 만일 없다면 httpd 와 2 진파일을 련결하며 그다음 자기의 조작체계(우리의 경우 Linux)를 선택한다. 여기서 리용할수 있는 파일들의 목록을 찾을것이다.

이러한 파일들을 훑어 볼 때 가장 가까운 판본은 처리소자에 의존한다. 례를 들어 i586 과 i686 명칭들은 제품이 Pentium 이나 Pentium Ⅱ 부류의 처리소자용으로 최량화콤파일된것이라는것을 가리킨다.

임의의 내리적재등록부안에서 README 파일들은 본문파일들인데 내리적재하기전에 그것들을 읽어 보는것이 좋다. GZ 파일들은 gzip 편의프로그람을 리용하여 압축된것들이다. ASC 파일들은 PGP 안전서명을 내포하고 있다. 일단 적재할 제품을 선택하면 림시위치에 파일을 전개한다. 현행등록부의 보조등록부안에 문서고를 전개하기 위하여 아래와 같은 지령을 리용한다.

$ tar xzvf apache_*file-name*

이 경우의 보조등록부는 apache_1.3.20 이라고 부르는데 이것은 2 진으로 적재할 때 리용할수 있는 Apache 의 판본에 따라서 서로 다를수 있다. 이 등록부안에 견본구성파일들과 HTML 형식의 문서와 함께 적재하려는 판본에 대한 완전한 원천코드가 있다.

2 진파일들을 설치하기 위해서는 전개된 문서고의 웃준위등록부로부터 설치스크립트 install-bindist.sh 를 실행해야 한다.

$ **install-bindist.sh**

이 지령은 기정에 의해 /usr/local/apache 안에 Apache 봉사기, 구성파일들과 Web 문서들을 설치한다. 이것은 Red Hat 에 따라 차이나는데 Apache 봉사기는 /usr/sbin 안에 있고 구성파일들은 /etc/httpd/conf 안에 있으며 Web 문서들을 위하여 /var/www/html 뿌리등록부를 만든다. 이 장에서는 만일 이 책에 포함된 CD-ROM 으로부터 Apache 를 설치한다면 이것을 볼수 있기때문에 Red Hat 기정이라고 가정한다.

주의

만일 Apache 의 다른 판본을 리용하고 있다면 등록부들은 차이날수 있다. find 나 locate 지령들을 리용하여 해당 등록부를 찾을수 있다

2진 Apache 배포물들은 포함된 원천코드로부터 응용프로그람을 콤파일하기 위한 완전한 지령들을 포함하고 있다. 그러므로 만일 적재된 2 진파일들이 기대한것처럼 동작하지 않는다(아마 Linux 의 판본안에서 서고들의 판본이 일치하지 않는것과 같은 문제들때문에)는것을 발견하면 그 지령들에 따라 가면서 자체로 Apache 2 진파일들을 만들어볼수도 있다.

Apache 구성

1.3.6 판 이전에는 Apache 는 세개의 기본 구성파일들 즉 httpd.conf, srm.conf 와 access.conf 을 리용하여 구성하였다. Apache 의 표준 Red Hat 설치에서 이러한 파일들은 /etc/httpd/conf/에서 보게 되는데 그러나 이 위치는 쉽게 변화(우리가 이 장에서 Apache Web 봉사기의 시동을 론의할 때 보게 되는것처럼)될수 있다. 1.3.6 판부터 이 세개의 파일들은 하나의 파일 httpd.conf 로 결합되였다. 이 파일은 모든 세개 파일의 목적들을 포함하고 있으며 내용은 본질에 있어서 원래 세개 파일들을 결합한것과 꼭 같다.

이 세개의 구성파일들의 역할들은 그리 명백하게 정의되지 않고 일부 중복이 있지만 기본적으로 아래와 같이 리용된다.

httpd.conf 이 파일은 봉사기가 시동될 때 그안에 모듈들을 적재하는것과 함께 봉사기에 의하여 리용되는 포구수와 같은 일반적인 설정들을 하는데 리용한다.이 파일은 또한 srm.conf 와 access.conf 파일들의 위치를 표시한다.

srm.conf 이 파일은 봉사기의 뿌리문서나무와 CGI 프로그람들에 관련된 규칙들과 같은 다른 일반적인 설정들을 한다.

access.conf 이 파일은 봉사기 혹은 특수한 등록부들에 대한 접근조종제한을 설정하는데 리용될수 있다.

어떤 Web 주인이 열린 Web 봉사기우에서 Apache 를 실행하기전에 고려해야 하는 몇가지 주요설정들에 대하여 론의하자. 이러한 실례들에서는 Red Hat Linux 7.1 에 포함되여 있는 Apache 1.3.19 봉사기를 리용할것이다. Apache 의 이 판본은 하나의 /etc/httpd/conf/httpd.conf 구성파일을 리용한다.

파일의 이 파라메터들에 의해 기본봉사기가 HTTP 요구들에 대하여 어떻게 응답하는가하는 전체적인 Apache 봉사기의 특성과 Web 열람기를 가지고 있는 누구에게나 다 명백히 개별적인 Web 싸이트들처럼 보이는 임의의 가상주콤퓨터들의 응답을 지적할수 있다.

Apache 구성파일

/etc/httpd/conf 보조등록부가 원래 세개의 구성파일들을 모두 포함하고 있지만 반드시 구성해야 하는것은 오직 httpd.conf 파일뿐이다. 나머지 CONF 파일들은 호환성을 맞추기 위한것이다. httpd.conf 파일의 구조는 매우 간단하다.

아래에 기정 httpd.conf 파일을 제시한다.

```
##
## httpd.conf - Apache HTTP server configuration file
```

```
##
#
# Based upon the NCSA server configuration files originally by Rob Mc Cool.
#
# This is the main Apache server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://www.apache.org/docs/> for detailed information about
  the directives.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are uns
# ure consult the online docs. You have been warned.
#
# After this file is processed, the server will look for and process
# /usr/conf/srm.conf and then /usr/conf/access.conf
# unless you have overridden these with ResourceConfig and/or
# AccessConfig directives here.
#
# The configuration directives are grouped into three basic sections:
# 1. Directives that control the operation of the Apache server process
#   as a whole (the 'global environment').
# 2. Directives that define the parameters of the 'main' or 'default' server,
#   which responds to requests that aren't handled by a virtual host.
#   These directives also provide default values for the settings
#   of all virtual hosts.
# 3. Settings for virtual hosts, which allow Web requests to be sent to
#   different IP addresses or hostnames and have them handled by the
#   same Apache server process.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for Win32),
# the server will use that explicit path. If the filenames do ' *not* begin
# with "/", the value of ServerRoot is prepended - so "logs/foo.log"
# with ServerRoot set to "/usr/local/apache" will be interpreted by the
# server as "/usr/local/apache/logs/foo.log".
#
### Section 1: Global Environment
#
# The directives in this section affect the overall operation of Apache,
# such as the number of concurrent requests it can handle or where it
# can find its configuration files.
#
```

```
# ServerType is either inetd, or standalone. Inetd mode 1s only supported
on Unix platforms.
#
ServerType standalone
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# NOTE! If you intend to place this on an NFS (or otherwise network)
# mounted filesystem then please read the LockFile documentation
#(availableat<URL:http://www.apache.org/docs/mod/core.htm1#1ockf
# i1e>); you will save yourself a lot of trouble.
#
# Do NOT add a slash at the end of the directory path.
#
ServerRoot "/etc/httpd"
#
# The LockFile directive sets the path to the lockfile used when Apache
# is compiled with either USE_FCNTL_SERIALIZED_ACCEPT or
# USE_FLOCK_SERIALIZED_ACCEPT. This directive should normally be
# left at its default value. The main reason for changing it is if the logs
# directory is NFS mounted, since the lockfile MUST BE STORED ON
# A LOCAL DISK. The PID of the main server process is automatically
# appended to the filename.
#
LockFile /var/lock/httpd.lock
#
# PidFile: The file in which the server should record its process
# identification number when it starts.
#
PidFile /var/run/httpd.pid
#
# ScoreBoardFile: File used to store internal server process information.
# Not all architectures require this. But if yours does (you'll know because
# this file will be created when you run Apache) then you *-must* ensure
# that no two invocations of Apache share the same scoreboard file.
#
ScoreBoardFi1e /var/run/httpd.scoreboard
#
# In the standard configuration, the server will process this file,
# srm.conf, and access.conf in that order. The latter two files are
# now distributed empty, as it is recommended that all directives
```

```
# be kept in a single file for simplicity. The commented-out values
# below are the built-in defaults. You can have the server ignore
# these files altogether by using "/dev/null" (for Unix) or
# "nu1" (for Win32) for the arguments to the directives.
#
#ResourceConfig conf/srm.conf
#AccessConfig conf/access.conf
#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300
#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On
#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100
#
# KeepAliveTimeout: Number of seconds to wait for the next request from
# the same client on the same connection.
#
KeepAliveTimeout 15
#
# Server-pool size regulation. Rather than making you guess how many
# server processes you need, Apache dynamically adapts to the load it
# sees — that is, it tries to maintain enough server processes to
# handle the current load, plus a few spare servers to handle transient
# load spikes (e.g., multiple simultaneous requests from a single
# Netscape browser).
#
#It does this by periodically checking how many servers are waiting
# for a request. If there are fewer than MinSpareServers, it creates
# a new spare. If there are more than MaxSpareServers, some of the
# spares die off. The default values are probably OK for most sites.
#
MinSpareServers 5
MaxSpareServers 20
```

```
#
# Number of servers to start initially — should be a reasonable ballpark
# figure.
#
StartServers 8
#
# Limit on total number of servers running, i.e., limit on the number
# of clients who can simultaneously connect — if this limit is ever
# reached, clients will be LOCKED OUT, so it should NOT BE SET TOO
# LOW. It is intended mainly as a brake to keep a runaway server from
# taking the system with it as it spirals down...
#
MaxClients 150
#
# MaxRequestsPerChild: the number of requests each child process is
# allowed to process before the child dies. The child will exit so
# as to avoid problems after prolonged use when Apache (and maybe the
# libraries it uses) leak memory or other resources. On most systems, this
# isn't really needed, but a few (such as Solaris) do have notable leaks
# in the libraries. For these platforms, set to something like 10000
# or so; a setting of 0 means unlimited.
#
# NOTE: This value does not include keepalive requests after the initial
#    request per connection. For example, if a child process handles
#    an initial request and 10 subsequent "keptalive" requests, it
#    would only count as 1 request towards this limit.
#
MaxRequestsPerChild 100
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, in addition to the default. See also the <VirtualHost>
# directive.
#
#Listen 3000
#Listen 12.34.56.78:80
Listen 80
#
# BindAddress: You can support virtual hosts with this option. This direc
# tive is used to tell the server which IP address to listen to. It can either
# contain "*", an IP address, or a fully qualified Internet domain name.
# See also the <VirtualHost> and Listen directives.
```

```
#
#BindAddress *
#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a
# DSO you have to place corresponding 'LoadModule' lines at this loca
# tion so the directives contained in it are actually available _before_ they
# are used. Please read the file README.DSO in the Apache 1.3 distri
# bution for more details about the DSO mechanism and run 'httpd -1'
# for the list of already built-in (statically linked and thus always available)
modules in your httpd binary.
#
# Note: The order is which modules are loaded is important. Don't change
# the order below without expert advice.
#
# Example:
# LoadModule foo_modu1e modules/mod_foo. so
#LoadModu1e mmap_static_modu1e modules/mod_mmap_static.so
LoadModule vhost_a1ias_modu1e modu1es/mod_vhost_a1ias.so
LoadModule env_modu1e    modules/mod_env.so
LoadModule config_1og_modu1e modules/mod_1og_config.so
LoadModule agent_1og_modu1e modules/mod_1og_agent.so
LoadModule referer_1og_modu1e modul es/mod_1og_referer. so
#LoadModu1e mime_magic_modu1e modules/mod_mime_mag1c. so
LoadModule mime_modu1e    modu1es/mod_mime.so
LoadModule negotiation_modu1e modu1es/mod_negotiation.so
LoadModule status_modu1e    modules/mod_status.so
LoadModule info_modu1e     modules/mod_info.so
LoadModule includes_module   modu1es/mod_inc1ude.so
LoadModule autoindex_modu1e modu1es/mod_autcindex.so
LoadModule dir_modu1e       modu1es/mod_dir.so
LoadModule cgi_modu1e       modu1es/mod_cgi .so
LoadModule asis_modu1e     modu1es/mod_asis.so
LoadModule imap_modu1e     modules/mod_imap.so
LoadModule act1on_modu1e    modu1es/mod_actions.so
#LoadModu1e spe1ing_modu1e modu1es/mod_spe1ing.so
LoadModule userdir_modu1e  modu1es/mod_userdir.so
```

```
LoadModule a1ias_modu1e    modu1es/mod_a1ias.so
LoadModule rewrite_modu1e   modu1es/mod_rewrite.so
LoadModule access_modu1e    modu1es/mod__access.so
LoadModule auth_modu1e     modu1es/mod_auth.so
LoadModule anon_auth_modu1e modu1es/mod_auth_anon.so
LoadModule db_auth_modu1e   modu1es/mod_auth_db.so
#LoadModu1e digest_modu1e   modu1es/mod_digest.so
#LoadModu1e proxy_modu1e   modules/libproxy.so
#LoadModu1e cern_meta_modu1e modu1es/mod_cern_meta.so
LoadModule expires_modu1e    modu1es/mod_expires.so
LoadModule headers_modu1e    modu1es/mod_headers.so
#LoadModu1e usertrack_modu1e modu1es/mod_usertrack.so
#LoadModu1e examp1e_modu1e   modu1es/mod_examp1e.so
#LoadModu1e unique_id_modu1e modu1es/mod_unique_id.so
LoadModule setenvif_modu1e    modu1es/mod_setenvif.so
#LoadModu1e bandwidth_modu1e modu1es/mod_bandwidth.so
#LoadModu1e put_modu1e     modu1es/mod_put.so
<IfDefine HAVE_PERL>
LoadModule per1__modu1e     modules/libperl.so
</IfDef1ne>
<IfDefine HAVE_PHP>
LoadModule php_modu1e      modu1es/mod_php.so
</IfDefine>
<IfDefine HAVE_PHP3>
LoadModule php3_modu1e     modu1es/1ibphp3.so
</IfDefine>
<IfDefine HAVE_PHP4>
LoadModule php4_modu1e     modu1es/1ibphp4.so
</IfDefine>
<IfDefine HAVE_DAV>
LoadModule dav_modu1e      modules/libdav.so
</IfDefine>
<IfDefine HAVE_ROAMING>
LoadModule roami ng_modu1e    modules/mod_roaming.so
</IfDefine>
<IfDefine HAVE_SSL>
LoadModule ss1_modu1e      modules/libssl.so
</IfDefine>
```

```
# Reconstruction of the complete module list from all available modules
# (static and shared ones) to achieve correct module execution order.
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE UPDATE THIS, TOO]
ClearModuleList
#AddModu1e mod_mmap_static.c
AddModule mod_vhost_a1ias.c
AddModule mod_env.c
AddModule mod_1og_confi g.c
AddModule mod_1og_agent.c
AddModule mod_1og_referer.c
#AddModu1e mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
AddModule mod_status.c
AddModule mod_info.c
AddModule mod_i nclude.c
AddModule mod_autoindex.c
AddModule mod_dir.c
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_imap.c
AddModule mod_actions.c
#AddModu1e mod_spe1ing.c
AddModule mod_userdir.c
AddModule tnod_a1ias.c
AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
AddModule mod_auth_anon.c
AddModule mod_auth_db. c
#AddModu1e mod_digest.c
#AddModu1e mod_proxy.c
#AddModu1e mod_cern_meta.c
AddModule mod_expires.c
```

```
AddModule mod_headers.c
#AddModu1e mod_usertrack.c
#AddModu1e mod_examp1e.c
#AddModu1e mod_unique_id.c
AddModule mod_so.c
AddModule mod_setenvif.c
#AddModu1e mod_bandwidth.c
#AddModu1e mod_put.c
<IfDefine HAVE_PERL>
AddModule mod_per1.c
</IfDefine>
<IfDefine HAVE_PHP>
AddModule mod_php.c
</IfDefine>
<IfDefine HAVE_PHP3>
AddModule mod_php3.c
</IfDefine>
<IfDefine HAVE_PHP4>
AddModule mod_php4.c
</IfDefine>
<IfDefine HAVE_DAV>
AddModule mod_dav.c
</IfDefine>
<IfDefine HAVE_ROAMING>
AddModule mod_roaming. c
</IfDefine>
<IfDefine HAVE_SSL>
AddModule mod_ss1.c
</IfDefine>
#
# ExtendedStatus: controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information (ExtendedSta
# tus Off) when the "server-status" handler is called. The default is Off.
#
# ExtendedStatus On
### Section 2: 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition. These values also provide defaults for
# any <Virtua1Host> containers you may define later in the file.
```

```
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#
#
# If your ServerType directive (set earlier in the 'Global Environment'
# section) is set to "inetd", the next few directives don't have any
# effect since their settings are defined by the inetd configuration.
# Skip ahead to the ServerAdmin directive.
#
#
# Port: The port to which the standalone server listens. For
# ports < 1023, you will need httpd to be run as root initially.
#
Port 80
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd
# as. . On SCO (ODT 3) use "User nouser" and "Group nogroup".
# . On HPUX you may not be able to use shared memory as nobody,
# and the suggested workaround is to create a user www and use that
# user. NOTE that some kernels refuse to setgid(Group) or semctl(IPC_
# SET) when the value of (unsigned)Group is above 60000;
# don't use Group nobody on these systems!
#
User apache
Group apache
#
# ServerAdmin: Your address, where problems with the server should
# be e-mailed. This address appears on some server-generated pages,
# such as error documents.
#
ServerAdmin root@localhost
#
# ServerName: allows you to set a host name which is sent back to clients
# for your server if it's different than the one the program would get
# (i.e., use "www" instead of the host's real name).
#
# Note: You cannot just invent host names and hope they work. The
```

```
# name you define here must be a valid DNS name for your host. If you
# don't understand this, ask your network administrator.
# If your host doesn't have a registered DNS name, enter its IP address
# here. You will have to access it by its address (e.g., http://123.45.#67.89/)
anyway, and this will make redirections work in a sensible way.
#
#ServerName localhost
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/var/www/html"
#
# Each directory to which Apache has access, can be configured with
# respect to which services and features are allowed and/or disabled in
# that directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# permissions.
#
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>
#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#
#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "/var/www/html">
#
# This may also be "None", "All", or any combination of "Indexes",
# "Includes", " FollowSymLinks", "ExecCGI", or "MultiViews".
#
# Note that "MultiViews" must be named *explicitly* — "Options All"
# doesn't give it to you.
```

```
#
    Options Indexes Includes FoilowSymLinks
#
# This controls which options the .htaccess files in directories can
# override. Can also be "All", or any combination of "Options", "FileInfo",
# "AuthConfig", and "Limit"
#
    AllowOverride None
#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

#
# UserDir: The name of the directory which is appended onto a user's
# home directory if a ~user request is received.
#
UserDir pub1ic_htm1
#
# Control access to UserDir directories. The following is an example
# for a site where these directories are restricted to read-only.
#
#<Directory /home/*/pub1ic_htm1>
#  AllowOverride FileInfo AuthConfig Limit
#  Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#  <Limit GET POST OPTIONS PROPFIND>
#    Order allow,deny
#    Allow from all
#  </Limit>
# <Limit PUT DELETE PATCH PROPPATCH MKCOL COPY MOVE LOCK
    UNLOCK>
#    Order deny, allow
#    Deny from all
#  </Limit>
#</Directory>
#
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index. Separate multiple entries with spaces.
```

```
#
DirectoryIndex index.html index.htm index.shtml index.php index.php4
    index .php3 index.cgi
#
# AccessFileName: The name of the file to look for in each directory
# for access control information.
#
AccessFileName .htaccess
#
# The following lines prevent .htaccess files from being viewed by
# Web clients. Since .htaccess files often contain authorization
# information, access is disallowed for security reasons. Comment
# these lines out if you want Web visitors to see the contents of
# .htaccess files. If you change the AccessFileName directive above,
# be sure to make the corresponding changes here.
#
# Also, folks tend to use names such as .htpasswd for password
# files, so this will protect those as well.
#
<Fi1es ~"^\ .ht">
    Order allow,deny
    Deny from all
</Fi1es>
#
# CacheNegotiatedDocs: By default, Apache sends ''Pragma: no-cache"
# with each document that was negotiated on the basis of content. This
# asks proxy servers not to cache the document. Uncommenting the follo
# wing line disables this behavior, and proxies will be allowed to cache
# the documents.
#
#CacheNegotiatedDocs
#
# UseCanonicalName: (new for 1.3) With this setting turned on, whene
# ver Apache needs to construct a self-referencing URL (a URL that refe
# rs back to the server the response is coming from) it will use ServerName
# and Port to form a "canonical" name. With this setting off, Apache will
# use the hostname:port that the client supplied, when possible. This
# also affects SERVER_NAME and SERVER_PORT in CGI scripts.
#
UseCanonicalName On
#
```

```
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
TypesConfig /etc/mime.types
#
# DefaultType is the default MIME type the server will use for a document
# if it cannot otherwise determine one, such as from filename extensions.
# If your server contains mostly text or HTML documents, "text/plain"
# is a good value. If most of your content is binary, such as applications
# or images, you may want to use "application/octet-stream" instead to
# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain
#
# The mod_mime_magic module allows the server to use various hints from
# the contents of the file itself to determine its type. The MIMEMagicFi1e
# directive tells the module where the hint definitions are located.
# mod_mime_magic is not part of the default server (you have to add
# it yourself with a LoadModule [see the DSO paragraph in the 'Global
# Environment' section], or recompile the server and include mod_mime_
# magic as part of the configuration), so it's enclosed in an <IfModu1e>
container.
# This means that the MIMEMagicFi1e directive will only be processed
# if the module is part of the server.
#
<IfModule mod_mime_magic.c>
MIMEMagicFile /usr/share/magic
</IfModu1e>
#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off
#
# ErrorLog: The location of the error log file.
```

```
# If you do not specify an ErrorLog directive within a <Virtua1Host>
# container, error messages relating to that virtual host will be
# logged here. If you *do* define an error "logfile for a <Virtua1Host>
# container, that host's errors will be logged there and not here.
#
ErrorLog /var/1og/httpd/error_1og
#
# LogLevel: Control the number of messages logged to the error_1og.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn
#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %1 %u %t \ "%r \ " %>s %b \ "%{Referer}i \ "\ "%{User-Agent}i
\ ""
    combined
LogFormat "%h %1 %u %t \ "%r \ " %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <Virtua1Host>
# container, they will be logged here. Contrariwise, if you *do*
# define per-<Virtua1Host> access logfiles, transactions will be
# logged therein and *not* in this file.
#
CustomLog /var/1og/httpd/access_1og common
#
# If you would like to have agent and referer logfiles, uncomment the
# following directives.
#
#CustomLog /var/1og/httpd/referer_1og referer
#CustomLog /var/1og/httpd/agent_1og agent
#
# If you prefer a single logfile with access, agent, and referer information
# (Combined Logfile Format) you can use the following directive.
#
#CustomLog /var/1og/httpd/access_1og combined
#
```

```
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (error documents, FTP directory listin
# gs, mod_status and mod_1nfo output etc., but not CGI generated docum
# ents).Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of: On | Off ] EMail
#
ServerSignature On
#
# Aliases: Add here as many aliases as you need (with no limit). The
# format is Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL. So "/icons" isn't aliased in this
# example, only "/icons/"..
#
Alias /icons/ "/var/www/icons/"
<Directory "/var/www/icons">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
#
# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to
# the client. The same rules about trailing "/" apply to ScriptAlias direc
# tives as to Alias.
#
ScriptAlias /cgi-bin/ " /var/www/cgi-bin/"
#
# "/var/www/cgi-bin should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.
#
<Directory " /var/www/cgi-bin">
    AllowOverride None
    Options ExecCGI
    Order allow,deny
    Allow from all
</Directory>
```

```
#
# Redirect allows you to tell clients about documents which used to exist
# in your server' s namespace, but do not anymore. This allows you to
# tell the clients where to look for the relocated document.
# Format: Redirect old-URI new-URL
#
# Directives controlling the display of server-generated directory listings.
#
# FancyIndexing: whether you want fancy directory indexing or stan
# dard
#
IndexOptions FancyIndexing
#
# AddIcon* directives tell the server which icon to show for different
# files or filename extensions. These are only displayed for
# FancyIndexed directories.
#
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip
AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif)image/*
AddIconByType (SND,/icons/sound2. gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/wor1d2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
AddIcon /icons/layout.gif .html .shtml .htm .pdf
AddIcon /icons/text.gif .txt
AddIcon /icons/c.gif .c
AddIcon /icons/p.gif .pl .py
AddIcon /icons/f.gif .for
AddIcon /icons/dvi.gif .dvi
AddIcon /icons/uuencoded.gif .uu
AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
AddIcon /icons/tex.gif .tex
AddIcon /icons/bomb.gif core
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
```

```
AddIcon /icons/folder.gif ^^DI RECTORY^^
AddIcon /icons/blank. gif ^^BLANKICON^^
#
# DefaultIcon: which icon to show for files which do not have an icon
# explicitly set.
#
DefaultIcon /icons/unknown.gif
#
# AddDescription: allows you to place a short description after a file in
# server-generated indexes. These are only displayed for FancyIndexed
# directories.
# Format: AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz
#
# ReadmeName: the name of the README file the server will look for
# by default, and append to directory listings.
#
# HeaderName: the name of a file which should be prepended to
# directory indexes.
#
# The server will first look for name.html and include it if found.
# If name.html doesn't exist, the server will then look for name.txt
# and include it as plaintext if found.
#
ReadmeName README
HeaderName HEADER
#
# IndexIgnore: a set of filenames which directory indexing should ignore
# and not include in the listing. Shell-style wildcarding is permitted.
#
Indexignore .??* *~ *
# HEADER* README* RCS CVS *,v *,t
#
# AddEncoding: allows you to have certain browsers (Mosaic/X 2.1+) unc
# ompress information on the fly. Note: Not all browsers support this.
# Despite the name similarity, the following Add* directives have nothing
# to do with the Fancylndexing customization directives above.
#
```

```
AddEncoding x-compress Z
AddEncoding x-gzip gz tgz
#
# AddLanguage: allows you to specify the language of a document. You
# can then use content negotiation to give a browser a file in a language
# it can understand. Note that the suffix does not have to be the same
# as the language keyword — those with documents in Polish (whose
# net-standard language code is pl) may wish to use "AddLanguagepl .po"
# to avoid the ambiguity with the common suffix for perl scripts.
#
AddLanguage en .en
AddLanguage fr .fr
AddLanguage de .de
AddLanguage da .da
AddLanguage el .el
AddLanguage it .it
#
# LanguagePriority: allows you to give precedence to some languages
# in case of a tie during content negotiation.
# Just list the languages in decreasing order of preference.
#
LanguagePriority en fr de
#
# AddType: allows you to tweak mime.types without actually editing it,
# or to make certain files to be certain types.
#
# The following is for PHP4 (conficts with PHP/FI, below):
 <IfModu1e mod_php4.c>
AddType application/x-httpd-php .php4 .php3 .phtml .php
AddType application/x-httpd-php-source .phps
</IfModu1e>
# The following is for PHP3:
<IfModu1e mod_php3.c>
AddType app1ication/x-httpd-php3 .php3
AddType app1ication/x-httpd-php3-source .phps
</IfModu1e>

# The following is for PHP/FI (PHP2):
<IfModu1e mod_php.c>
AddType application/x-httpd-php .phtml
```

```
</IfModule>
AddType application/x-tar .tgz
#
# AddHandler: allows you to map certain file extensions to "handlers",
# actions unrelated to filetype. These can be either built into the server
# or added with the Action command (see below)
#
# If you want to use server side includes, or CGI outside
# ScriptAliased directories, uncomment the following lines.
#
# To use CGI scripts:
#
#AddHandler cgi-script .cgi
#
# To use server-parsed HTML files
#
AddType text/html .shtml
AddHandler server-parsed .shtml
#
# Uncomment the following line to enable Apache's send-asis HTTP file
# feature
#
#AddHandler send-as-is asis
#
# If you wish to use server-parsed imagemap files, use
#
AddHandler imap-file map
#
# To enable type maps, you might want to use
#
#AddHandler type-map var
#
# Action: lets you define media types that will execute a script whenever
# a matching file is called. This eliminates the need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location
#

#
# MetaDir: specifies the name of the directory in which Apache can find
```

```
# meta information files. These files contain additional HTTP headers
# to include when sending the document
#
#MetaDir .web
#
# MetaSuffix: specifies the file name suffix for the file containing the
# meta information.
#
#MetaSuffix .meta
#
# Customizable error response (Apache style)
#  these come in three flavors
#
#  1) plain text
#ErrorDocument 500 "The server made a boo boo.
#  n.b. the (") marks it as text, it does not get output
#
#  2) local redirects
#ErrorDocument 404 /missing.html
#  to redirect to local URL /missing.html
#ErrorDocument 404 /cgi-bin/missing_handler.pl
# N.B.: You can redirect to a script or a document using server-side-
# includes.
#
#  3) external redirects
#ErrorDocument 402 http://some.other_server.com/subscription_info.ht # ml
N.B.: Many of the environment variables associated with the original
# request will *not* be available to such a script.
#
# The following directives modify normal HTTP response behavior.
# The first directive disables keepalive for Netscape 2.x and browsers
# that spoof it. There are known problems with these browser implem
# entations. The second directive is for Microsoft Internet Explorer 4.0b2
# which has a broken HTTP/1.1 implementation and does not properly
# support keepalive when it is used on 301 or 302 (redirect) responses.
#
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4 \ .0b2;" nokeepalive downgrade-1.0
force-response-1.0
#
# The following directive disables HTTP/1.1 responses to browsers which
```

```
# are in violation of the HTTP/1.0 spec by not being able to grok a
# basic 1.1 response.
#
BrowserMatch "RealPlayer 4\ .0" force-response-1.0
BrowserMatch "Java/1\ .0" force-response-1.0
BrowserMatch "JDK/1\ .0" force-response-1.0
# If the perl module is installed, this will be enabled.
<IfModule mod_perl.c>
Alias /perl/ /var/www/perl/
<Location /perl>
    SetHandler perl-script
    PerlHandler Apache::Registry
    Options +ExecCGI
</Location>
</IfModule>
#
# Allow http put (such as Netscape Gold's publish feature)
# Use htpasswd to generate /etc/httpd/conf/passwd.
# You must unremark these two lines at the top of this file as well:
#LoadModule put_module        modules/mod_put.so
#AddModule mod_put.c
#
#Alias /upload /tmp
#<Location /upload>
#  EnablePut On
#  AuthType Basic
#  AuthName Temporary
#  AuthUserFite /etc/httpd/conf/passwd
#  EnableDelete Off
#  umask 007
#  <Limit PUT>
#  require valid-user
#  </Limit>
#</Location>
#
# Allow server status reports, with the URL of http://servername/ server
# -status Change the ".your_domain.com" to match your domain to en
# able.
#
#<Location /server-status>
```

```
#   SetHandler server-status
#   Order deny,allow
#   Deny from all
#   Allow from . your_domain. com
#</Location>
#
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Change the ".your_domain.com" to match your domain to enable.
#
#<Location /server-info>
#  SetHandler server-info
#  Order deny,allow
#  Deny from all
#  Allow from .your_domain. com
#</Location>

# Allow access to local system documentation from localhost
Alias /doc/ /usr/share/doc/
<Location /doc>
    order deny,allow
    deny from all
    allow from localhost
    Options Indexes FollowSymLinks
</Location>
#
# There have been reports of people trying to abuse an old bug from
# pre-1.1 days. This bug involved a CGI script distributed as a part of
# Apache. By uncommenting these lines you can redirect these attacks
# to a logging script on phf.apache.org. Or, you can record you're your
# self, using the script support/phf_abuse_1og.cgi.
#
#<Location /cgi-bin/phf*>
#   Deny from all
#   ErrorDocument 403 http://phf.apache.org/phf_abuse_1og.cgi
#</Location>
#
# Proxy Server directives. Uncomment the following lines to
# enable the proxy server:
#
#<IfModu1e mod_proxy.c>
```

```
#ProxyRequests On
#
#<Directory proxy:*>
#  Order deny,allow
#  Deny from all
#  Allow from .your_domain.com
#</Directory>
#
# Enable/disable the handling of HTTP/1.1 "Via:'' headers.
# ("Full" adds the server version; "Block" removes all outgoing Via: hea
# ders)Set to one of: Off | On | Full | Block
#
#ProxyVia On
#
# To enable the cache as well, edit and uncomment the following lines:
# (no cacheing without CacheRoot)
#
#CacheRoot "/var/cache/httpd"
#CacheSize 5
#CacheGcInterval 4
#CacheMaxExpire 24
#CacheLastModifiedFactor 0.1
#CacheDefaultExpire 1
#NoCache a_domain.com another_domain.edu joes.garage_sale.com
#</IfModule>
# End of proxy directives.
### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on
# your machine you can setup Virtual Host containers for them.
# Please see the documentation at <URL:http://www.apache.org /docs/
# vhosts/> for further details before you try to setup virtual hosts.
# You may use the command line option '-S' to verify your virtual host
# configuration.
#
# If you want to use name-based virtual hosts you need to define at
# least one IP address (and port number) for them.
#
#NameVirtualHost 12.34.56.78:80
#NameVirtualHost 12.34.56.78
#
```

```
# Virtual Host example:
# Almost any Apache directive may go into a VirtualHost container.
#
#<Virtua1Host ip.address.of.host.some_domain.com>
#    ServerAdmin webmaste r@host.some_domain.com
#    DocumentRoot /www/docs/host. some_domain.com
#    ServerName host.some_domain.com
#    ErrorLog logs/host.some_domain.com-error_1og
#    CustomLog logs/host.some_domain.com-access_log common
#</VirtualHost>
#<VirtualHost _defau1t_:*>
#</VirtualHost>
<IfDefine HAVE_SSL>
##
## SSL Virtual Host Context
##
# Apache will only listen on port 80 by default. Defining the virtual server
# (below) won't make it automatically listen on the virtual server's port.
Listen 443
<VirtualHost _defau1t_:443>
# General setup for the virtual host
DocumentRoot "/var/www/html"
#  SSL Engine Switch:
#  Enable/Disable SSL for this virtual host.
SSLEngine on
#  SSL Cipher Suite:
#  List the ciphers that the client is permitted to negotiate.
#  See the mod_ss1 documentation for a complete list.
# SSLCipherSuiteALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2
# :+EXP:+eNULL
#  Server Certificate:
#  Point SSLCertificateFile at a PEM encoded certificate. If
#  the certificate is encrypted, then you will be prompted for a
#  pass phrase. Note that a kill -HUP will prompt again. A test
#  certificate can be generated with 'make certificate' under
#  built time. Keep in mind that if you've both a RSA and a DSA
#  certificate you can configure both in parallel (to also allow
#  the use of DSA ciphers, etc.)
SSLCertificateFile /etc/httpd/conf/ssl.crt/server.crt
#SSLCertificateFi1e /etc/httpd/conf/ssl.crt/server-dsa.crt
#  Server Private Key:
```

```
#  If the key is not combined with the certificate, use this
#  directive to point at the key file. Keep in mind that if
#  you've both a RSA and a DSA private key you can configure
#  both in parallel (to also allow the use of DSA ciphers, etc.)
SSLCertificateKeyFile /etc/httpd/conf/ssl.key/server.key
#SSLCertificateKeyF1le /etc/httpd/conf/ssl.key/server-dsa.key
#  Server Certificate Chain:
#  Point SSLCertificateChainFile at a file containing the
#  concatenation of PEM encoded CA certificates which form the
#  certificate chain for the server certificate. Alternatively
# the referenced file can be the same as SSLCertificateFile
# when the CA certificates are directly appended to the server
# certificate for convenience.
#SSLCertificateChainFi1e /etc/httpd/conf/ssl.crt/ca.crt
# Certificate Authority (CA):
# Set the CA certificate verification path where to find CA
# certificates for client authentication or alternatively one
# huge file containing a11 of them (file must be PEM encoded)
# Note: Inside SSLCACertificatePath you need hash symlinks
# to point to the certificate files. Use the provided
# Makefile to update the hash symlinks after changes.
#SSLCACertificatePath /etc/httpd/conf/ssl.crt
#SSLCACertificateFi1e /etc/httpd/conf/ssl.crt/ca-bundle.crt
# Certificate Revocation Lists (CRL):
# Set the CA revocation path where to find CA CRLs for client
# authentication or alternatively one huge file containing all
# of them (file must be PEM encoded)
# Note: Inside SSLCARevocationPath you need hash symlinks
# to point to the certificate files. Use the provided
# Makefile to update the hash symlinks after changes.
#SSLCARevocationPath /etc/httpd/conf/ssl.crl
#SSLCARevocationFi1e /etc/httpd/conf/ssl.crl/ca-bundle.crl
# Client Authentication (Type):
# Client certificate verification type and depth. Types are
# none, optional, require and optional_no_ca. Depth is a
# number which specifies how deeply to verify the certificate
# issuer chain before deciding the certificate is not valid.
# SSLVerifyClient require
# SSLVerifyDepth 10
# Access Control:
#With SSLRequire you can do per-directory access control based
# on arbitrary complex boolean expressions containing server
```

```
# variable checks and other lookup directives. The syntax is a
# mixture between C and Perl. See the mod_ss1 documentation
# for more details.
#<Location />
# SSLRequire ( %{SSL_CIPHER} !~ m/^EXPINULL)-/ \
# and %{SSL_CLIENT_S_DN_0} eq "Snake Oil, Ltd." \
# and %{SSL_CLIENT_S_DN_OU} in ("Staff", "CA", "Dev")\
# and %{TIME_WDAY} >= 1 and %(TIME_WDAY) <= 5\
#and %{TIME_HOUR} >= 8 and %(TIME_HOUR) <= 20   ) \
# or %{REMOTE_ADDR} =~ m/^192\ .76\ .162\ .[0-9]+$/
#</Location>
# SSL Engine Options:
# Set various options for the SSL engine.
# o FakeBasicAuth:
#Translate the client X.509 into a Basic Authorisation. This means
# that:the standard Auth/DBMAuth methods can be used for access
# control. The user name is the 'one line' version of the client's
#X.509 certificate.
# Note that no password is obtained from the user. Every entry in
# the user file needs this password: 'xxj31ZMTZzkVA'.
# o ExportCertData:
# This exports two additional environment variables: SSL_CLIENT_
# CERT and SSL_SERVER_CERT. These contain the PEM-encoded
# certificates of the server (always existing) and the client (only
# existing when client authentication is used). This can be used
#to import the certificates into CGI scripts.
# o StdEnvVars:
#This exports the standard SSL/TLS related 'SSL_*' environment
# variables Per default this exportation is switched off for performa
# nce reasons,because the extraction step is an expensive operation
# and is usually useless for serving static content. So one usually
# enables the exportation for CGI and SSI requests only.
# o CompatEnvVars:
# This exports obsolete environment variables for backward compatibility
# to Apache-SSL 1.x, mod_ss1 2.0.X, Sioux 1.0 and Stronghold 2.x.
# Use this to provide compatibility to existing CGI scripts.
# o StrictRequire:
#This denies access when "SSLRequireSSL" or "SSLRequire" appl
```

```
#ied even under a "Satisfy any" situation, i.e. when it applies acce
# ss is denied and no other module can change it.
#o OptRenegotiate:
# This enables optimized SSL connection renegotiation handling
#when SSL directives are used in per-directory context.
#SSLOptions +FakeBasicAuth +ExportCertData +CompatEnvVars +Stric
# tRequire
<Files ~ "\.(cgi|shtml)$">
    SSLOptions +StdEnvVars
</Files>
<Directory "/var/www/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>
#  Notice: Most problems of broken clients are also related to the HTTP
#  keep-alive facility, so you usually additionally want to disable
#  keep-alive for those clients, too. Use variable "nokeepalive" for this.
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown
#  Per-Server Logging:
#  The home of a custom SSL log file. Use this when you want a
#  compact non-error SSL logfile on a virtual host basis.
CustomLog /var/log/httpd/ssl_request_log \
    "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \ "%r\ " %b"
</VirtualHost>
</IfDefine>
```

우선 주목해야 할것은 이 파일이 설명문들과 실지 구성항목들로 이루어 져 있다는 것이다. 설명들에는 하쉬표식(#)이 붙어 있다. 구성항목들은 항목값들에 따르는 항목이름으로 구성된다.

httpd.conf 파일안에는 세가지 기본토막들: 즉 전체적인 환경, 기본봉사기구성, 가상주콤퓨터들이 들어 있다. 매 토막을 자세히 보자.

전체적인 환경

이 토막에서의 지령들과 모듈들은 전체적인 Apache 봉사기프로쎄스들의 동작을 조종한다. 여기에서 본것은 모든 Apache 주콤퓨터들에 적용되여 그것이 주봉사기인가 아니면 다음토막에서 론의되는 가상주콤퓨터인가를 알아 볼수 있게 한다.

ServerType

배워야 하는 첫번째 지령은 ServerType이다. Apache봉사기는 두가지 방식 즉 접속을 접수하는 자체프로쎄스들을 가진 standalone방식과 망에 대한 토막에서 보게 되는 inetd방식으로 실행할수 있다. inetd방식에서는 inetd가 접속들을 기다리다가 접속들이 요구되였을 때 Apache프로쎄스를 시작한다. 만일 Web봉사기를 처음 실행해 볼 때에는 inetd 를 잘 알지 못하면 standalone 방식으로 Apache 를 실행하는것이 가장 좋은데

Unix 봉사기를 리용할 때 이 방식으로 봉사기를 실행해야 할 리유들이 있다.

ServerRoot

ServerRoot 는 구성파일들과 봉사기에 대한 등록들이 들어 있는 기본등록부를 가리키는데 리용된다. Red Hat 는 이것을 /etc/httpd 등록부안에 설치한다. 이 등록부안에 세개의 구성파일들을 포함하고 있는 conf 등록부와 /var/log/httpd 에 련결된 logs 등록부가 있다.

사용하고 있는 관리방식에 적합하게 자기의 체계의 임의의 곳에 등록부를 자유롭게 이동할수 있다. 그러나 보통 Linux 배포물이 배치한 그대로 그것들을 놔두는것이 가장 편리하다.

Loading Modules

첫번째 기본토막은 LoadModule 지령들의 렬과 그뒤의 AddModule 지령렬들로써 구성된다. Apache 의 기능은 모듈들을 리용하여 창조되고 확장된다. Apache 의 이전 판본들에서는 이러한 모듈들이 실행가능한 Apache httpd 파일에 결합되였었다. Apache 1.3 판본부터는 봉사기가 기동하는 때에 적재하는 모듈들을 생성할수 있다.

이러한 모듈들의 거의 대부분이 지령들을 통하여 구성되기때문에 모든 적절한 모듈들을 다른 지령들이 httpd.conf 파일안에서 발생되기전에 적재하는것이 중요하다.

가장 초보적인 설치를 하려면 기정모듈목록을 쓰는것이 좋으며 그것을 변경시킬 필요가 없다. 만일 자기의 Web 봉사기의 기능을 확장하거나 변경시키려 할 때 적재가능한 모듈들을 추가하고 LoadModule 와 AddModule 지령들을 리용하는데 이에 대한 구체적인 정보를 얻자면 httpd.apache.org 에서 Apache 직결봉사문서를 참고해야 한다.

기본봉사기구성

Apache 봉사기우에 가상주콤퓨터들을 임의의 수만큼 설정할수 있다. 그러나 모든 Apache 봉사기는 기본주콤퓨터를 요구한다. 기본봉사기토막안에서의 지령들은 그 어느 가상주콤퓨터들에 한정되지 않는 임의의 지령들과 마찬가지로 기본주콤퓨터를 위한 기정파라메터들을 결정한다.

포구

다음으로 중요한 항목은 Port 항목이다. 기정으로 Web 봉사는 포구 80 에서 실행되도록 되여 있다. 열람기가 포구를 밝히지 않고 URL 을 요구했을 때에는 URL 이 포구 80 으로 봉사기로부터 요구하고 있다고 가정한다. 만일 고수준 혹은 중수준의 안전방화벽을 실행하고 있었다면 제 31 장《보안과 값 눅은 경로기로서의 Red Hat Linux 7.1》에서 론의된것처럼 WWW 자료가 들어 오도록 전용화되였는가를 확인한다.

포구 1024 이하에서 실행하고 있는 프로쎄스들의 약점으로 인하여 만일 인터네트우에서 공개적으로 쓸수 있게 만들어 지지 않는 Web 봉사기를 실행하면서 이러한 포구들과 관련된 위험성을 피하려고 한다면 포구 1024 이상의 포구에서 봉사기를 실행하려 할수 있다. 이러한 형태의 Web 봉사기들을 위해서 가장 많이 쓰이는 포구들은 8000 과 8080 이다. 그러나 임의의 자유포구를 선택할수도 있다. 만일 높은 안전성을 가진 방화벽을 실행하고 있다면 선택한 포구에 대한 WWW 자료입구를 확인한다. 제 31 장에서 서술한

lokkit 편의프로그람을 가지고 방화벽을 전용화하였을 때 포구 8080 을 리용하려고 한다면 전용화토막안의 Other Ports 본문칸안에 **8080:udp,8080:tcp** 라고 기입한다.

사용자와 그룹

사용자와 그룹 항목들은 체계의 안전성에 큰 영향을 미치기때문에 중요하면서도 위험한 항목들이다. 일반적으로 httpd 프로쎄스는 뿌리회계에 의하여 발화되지만 이 프로쎄스는 접속들을 접수하지 않는다. 오히려 이 프로쎄스들은 그것들의 구성지령에 의하여 지적되는 사용자와 그룹(이름 혹은 ID 번호에 의하여 지적된다.)과 같은 하나이상의 자식프로쎄스들을 발생시킨다.

이것은 CGI 프로그람들을 포함하여 Web 봉사기에 의하여 방화되는 모든 프로쎄스들이 다른 한편으로는 root 로서 실행되기때문에 수행된다. 이것은 특히 서툴게 작성된 CGI 스크립트가 큰 안전구멍을 남겨 놓는다면 커다란 안전위험성을 조장시키게 된다. 제한된 능력을 가진 사용자의 자격으로 Web 봉사기와 그의 관련자식들을 실행하는 방법으로 이러한 안전구멍들을 제한할수 있다.

표준조작은 apache 혹은 #-1 의 그룹을 가진 apache 로써 Web 봉사기를 실행하는것이다. 일반적으로 사용자 apache 와 그룹 apache 를 리용하는데 왜냐하면 그것들이 구성파일을 읽을 때 쉽게 식별될수 있기때문이다. 이 사용자는 체계우에서 제한된 권한들을 가지고 있으므로 만약 CGI 스크립트를 통하여 능력 있는 해커들이 체계안으로 침입하는 경우에도 파괴크기를 제한할수 있다.

봉사기관리

이 지령은 Web 싸이트 관리기의 전자우편주소를 가리킨다. 페지를 발견하지 못했다는것을 지시해 주는 등의 자동적인 오유통보문을 봉사기가 발생시키면 일반적으로 이 전자우편주소가 문제를 보고하기 위해 그 다음 누구와 접촉해야 하는가를 가리키고 있는 페지에 추가된다.이 주소가 유효하며 봉사기관리기의 정확한 주소라는것을 확인해야 한다.

봉사기이름

ServerName 구성지령은 자기가 요구한 페지들을 가지고 있는 의뢰기들에 되돌려지는 주콤퓨터이름을 나타낸다. 이것은 자기의 망우에서 DNS 레코드 혹은 주콤퓨터표안에서 식별되는것으로써 자기의 Web 봉사기에 대한 적당한 이름들중의 하나이다. 인트라네트환경안에서 NIS 가 리용하고 있는 주콤퓨터표들안에 혹은 자기의 싸이트를 위한 DNS 봉사기안에 주콤퓨터이름을 정의할수 있다. 인터네트 Web 봉사기에 대하여 사용자가 여기서 지적하는 이름이 사용자령역의 DNS 레코드안에서 유효한 이름인가를 확인해야 한다. 만일 확인할수 없다면 자기의 령역이름레코드들을 관리하는 관리기에게 의뢰해야 한다.

실례파일의 경우에 ServerName 은 localhost 이다. 만일 자체의 령역을 가지고 있다면 이것을 mommabears.com 과 같은 규칙적인 인터네트령역이름으로 변화될수 있다.

문서뿌리

DocumentRoot 구성항목은 HTML 파일용뿌리등록부가 어디에 위치하는가를 가리킨다. Red Hat Linux 7.1 에서는 보통 기정으로 /var/www/html 로 설정한다. 만일 ServerName 이 mommabears.com 으로 설정되였다면 열람기를 가지고 있는 어떤 사람

은 URL http://www.mommabears.com/file.html 을 리용하여 /var/www/html /file.html 을 호출할수 있어야 한다.

HTML 등록부의 허가

아래의 코드는 HTML 문서뿌리등록부를 취급하고 있는 실례 httpd.conf 파일의 일부분을 주해를 빼고 준것이다.

```
<Directory /var/www/htm1>
Options Indexes Includes FollowSymLinks
AllowOverride None
order allow,deny
allow from all
</Directory>
```

항목은 <Directory>와 </Directory>꼬리표들로 열리고 닫기는 안에 필요한 지령들을 포함시키는 구조로 되여 있다. 열기꼬리표는 꼬리표들사이에 있는 모든 지령들이 적용되여야 하는 등록부를 지적하는데 실례의 경우에는 /var/www/html 이다.

<Directory>와 </Directory>꼬리표들안에 4 가지 지령들이 제시되였다. Options 지령으로는 특수한 작용들이 등록부와 자기 보조등록부안에 포함된 파일들에서 일어 날수 있는가를 지시한다. 가능한 값들은 None, All, Indexes, Includes, FollowSymLinks, ExecCGI, MultiViews 이다. HTML 등록부들에 대하여 일반적으로 사용되는 값들은 표준 HTML 파일들과 화상들만 싸이트우에 나타날 때에는 None 이고 봉사기 해석형 HTML 파일들을 가능하게 하려 할 때는 Includes 이다(일부 봉사기의 해석형 HTML 파일들은 파일안에 다른 파일들을 포함시킬수 있는 능력을 제공하는데 Includes 값은 이 작용이 일어 나도록 한다.).

다음지령은 AllowOverride 항목인데 그것은 국부 .htaccess 파일이 대역 httpd.conf 파일안에 있는 항목들을 어느정도 무시해야 하는가를 가리킨다. 가능한 값들은 None, All, Options, FileInfo, AuthConfig, Limit 이다. 실례로 Options 값은 .htaccess 파일이 Options 지령을 무시하도록 한다. Web 주인이 모든 내용을 원만히 조종할수 있는 봉사기들에서는 보통 무시할수 있는 완전한 권리를 허용하는것이 제일 쉽다. 자기자체의 등록부들의 내용을 조종하는 다중사용자들을 가지고 있는 봉사기를 실행할 때에는 대역적인 접근구성을 무시하려고 하는 사용자들의 능력을 제한하는것이 좋을수도 있다.

마지막으로 order 와 allow 지령들은 등록부안에서 페지들에 누가 접근하는가를 조종하는데 함께 쓰인다. 여기서 order allow,deny 는 처음 allow 지령을 리용해야 하는데 만일 그 지령이 사용자가 질문파일을 얻지 못하도록 하였다면 그다음 deny 지령이 적용되여야 한다는것을 가리킨다.

보통 접근조종을 적용하지 않고 있을 때 order allow,deny 를 리용하려고 할것이다. 접근조종을 하려고 시도할 때(후에 보게 되는것처럼 접근조종을 가진 작은 표본 Web 싸이트를 구축할 때) order deny,allow 가 가장 좋은 항목이다.

order 지령에 뒤따르는것이 allow 지령인데 그것은 모든 사용자들의 접근을 허가해야 한다는것을 나타낸다. 만일 사용자를 어떻게 거절하는가를 완전히 리해하려고 한다면 http://httpd.apache.org에서 직결로 리용할수 있는 Apache 문서를 읽어야 한다.

사용자등록부

UserDir 는 Web 주인이 체계우의 매 사용자들이 자기의 홈등록부안에 있는 보조등록부를 통하여 관리하는 자기자체의 개인적인 Web 싸이트를 가지게 하려고 할 때 쓸모있는 지령이다. 이 지령은 자기의 Web 등록부로 고려되는 자기의 홈등록부안에 있는 보조등록부의 이름을 표시한다.

표준은 public_html 을 리용하는것이다. 그래서 만일 사용자 testuser 가 /home/testuser/public_html 등록부를 가지고 있다면 이 등록부는 URL http://servername/~testuser를 써서 Web를 통하여 호출될수 있다.

등록부색인

DirectoryIndex 는 어느 파일들이 기정파일들로 간주되여야 하는가를 나타내는 중요한 지령이다. 이것은 http://www.mommabears.com과 같은 URL 에 의해서 정확한 파일을 호출하는것이 어느 정도 가능하게 한다.

실례를 들어 우의 실례 httpd.conf 파일안에서 등록색인을 위하여 주어 진 7 개의 입구점 즉 index.html, index.htm, index.shtml, index.php, index.php4, index.php3, index.cgi 가 있다. 이것은 파일이름은 지적하지 않고 오직 등록부만 주어진 어떤 URL 에 대하여 봉사기가 처음에 지적된 등록부로부터 index.shtml 을 돌려 주려고 시도하게 된다는것을 의미한다. 만일 그 파일이 없다면 index.htm 이 되돌려 지고 또 그 파일이 없으면 index.shtml 를 다시 의뢰기에 보낸다. 만일 적합한 파일을 찾지 못했다면 봉사기는 다른 구성선택항목들에 기초하여 등록부목록 혹은 오유통보문을 되돌려 준다.

접근파일이름

AccessFileName 은 주어진 등록부에 대한 접근조종정보를 포함하는 파일이름을 지적하는데 리용된다. 매 home 등록부안에서 httpd.conf 파일 혹은 .htaccess 구성파일들안에 접근조종정보를 보관하는것이 가능하다.

실례파일은 등록부안에 이름이 .htaccess 인 파일이 존재하면 그 파일이 등록부에 대한 접근조종정보를 포함한다는것을 가리킨다.

스크립트별명

CGI 프로그람들과 스크립트들을 보관하고 있는 등록부를 설치할수 있도록 하기 위하여 ScriptAlias 지령을 설정하는것이 중요하다. ScriptAlias 는 어느 등록부가 CGI 스크립트를 위하여 리용되였으며 그 등록부에 대한 URL 이 무엇인가를 지적한다. 다른 권한(다음 소절의 Add Handler 지령의 론의에서 보게 되는것처럼) 에 의해서 파일확장자에 기초한 CGI 프로그람을 실행하도록 지적되지 않는 한 이 등록부안에 있는 파일들은 자동적으로 CGI 프로그람들로 취급된다.

례컨대 실례파일은 아래의 지령을 리용했다.

```
ScriptAlias /cgi-bin/ " /var/www/cgi-bin"
```

이 행은 URL http://www.mommabears.com//cgi-bin/이 /var/www/ cgi-bin /을 가리킨다는것을 나타낸다. 이 등록부안에서 파일들은 CGI 스크립트들로 간주되며 봉

사기들은 의뢰기요구에 직접 그것들을 되돌려 보내는것이 아니라 그 파일들을 실행하려하게 된다.

CGI 등록부의 허가

앞에서 서술된 HTML 등록부항목과 비교해 볼 때 CGI 등록부항목은 약간한 차이를 가지고 있다.

```
<Directory /var/www/cgi-bin>
AllowOverride None
Options ExecCGI
Order allow,deny
Allow from all
</Directory>
```

모든 무효가 허가되지 않았다는데 주목한다. 이것은 일반적으로 현명한 생각인데 왜냐하면 CGI 는 잘 구성된 체계우에서조차 안전구멍으로 되며 그것은 CGI 등록부들이 가지고 있는 모든 가능한 안전성결함들을 막기 위해 조심해야 하기때문이다.

추가조종기와 추가형태

AddHanlder 와 Addtype 지령들은 실제로 함께 고려되여야 할 필요가 있다. AddHanlder 는 지적된 확장자를 가진 파일들이 특정의 동작에 대응되게 하는데 그 동작은 (CGI 프로그람들을 실행하는것과 같이)봉사기안에서 내장된 동작이 될수도 있고 혹은 봉사기밖에서 특수한 프로그람을 일반적으로 실행하고 그 프로그람을 통하여 문제의 파일을 처리하는 외부적인 동작이 될수도 있다.

Addtype 는 특수한 확장자를 위한 새로운 MIME 형을 창조한다. MIME 형들은 의뢰기가 파일을 어떻게 조종하는가를 통지하는데서 중요하다. 실례를 들어 파일이 MIME 형의 text/plain 에 의하여 열람기에 넘어 갔다면 열람기는 그것이 포함하고 있는 임의의 HTML 코드파일을 해석하려고 하지 않고 MIME 형의 text/html 이 열람기가 그 파일을 HTML 파일로 접수하도록 처리하게 한다.

이 지령들의 주요 두가지 사용은 지적된 CGI 스크립트등록부밖에서 CGI 스크립트들이 실행되도록 하는것과 봉사기의 해석형 HTML 파일을 가능하게 하여 HTML 파일안에 들어 있는 특수한 꼬리표들이 봉사기가 의뢰기에 페지를 돌려 보내기전에 봉사기에 의하여 처리되도록 하는것이다.

CGI 스크립트의 허가

AddHandler 지령을 리용하여 지적된 CGI 등록부밖에서 CGI 를 처리하도록 할수 있다. 그를 위해 httpd.conf 파일에 다음의 지령을 추가한다.

```
AddHandler cgi-script .cgi
```

이것은 지적된 CGI 등록부밖에서 발견된 확장자 .cgi 를 가진 임의의 파일이 CGI 프로그람으로서 취급되여야 하며 그런 식으로 조종되여야 한다는것을 나타낸다. 이 지령이 없으면 CGI 등록부밖에서 발견한 CGI 스크립트는 CGI 프로그람으로서 처리되지 못하며 대신 그 파일의 내용이 단순히 의뢰기에 보내여진다. 일반적으로 이것은 사용자가 스크립트를 실행한 결과를 보기 보다는 오히려 스크립트의 실지 프로그람코드를 보게 된다

는것을 의미한다.

봉사기가 해석한 HTML의 허가

AddHandler와 AddType지령들은 둘 다 봉사기의 해석형HTML을 가능하게 하는데 리용된다. Apache의 일반사용은 다음과 같다.

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

여기서 AddType은 봉사기가 해석한 HTML파일(확장자 .shtml을 가진)의 결과들을 의뢰기열람기에 의하여 HTML로서 보게 하며 그와 같이 표시되도록 담보한다.

AddHandler 행은 확장자 .shtml 을 가진 파일들이 봉사기의 server-parsed 작용에 의하여 조종되여야 한다는것을 가리킨다. 이것은 SHTML 파일에 대하여 봉사기가 해석한 HTML이 효과를 가지게 한다.

Web봉사기관리

자체의 Web 봉사기를 시동 및 정지시키는것외에 일단 자체의 Web 봉사기를 설치하고 실행할 때 봉사기가 아무일없이 계속 실행되는가를 때때로 확인해야 하는 몇가지 과제들이 있다. 이러한 단계들은 사용자들과 그룹들의 만들기와 삭제,접근조종에 의한 등록부보호,그것들이 계속 쓸모있게 되도록 하기 위한 봉사기등록관리를 포함한다.

Apache의 시동과 정지

Red Hat를 설치할 때 Apache를 설치했다면 기동파일들은 체계시동시에 Apache를 시동하게 설치되여 있다. 이것은 /etc/rc.d/init.d/httpd 파일안에서 나타난다. 이 파일은 두개의 가능한 인수 즉 start 와 stop 을 가지고 실행할수 있는 스크립트이다. 만일 Red hat설치판에 포함된 Apache의 판본을 리용하는데 구성파일들의 위치를 옮기려 하지 않는다면 자기의 Web봉사기를 수동으로 시동시키고 정지시킬수 있는데

```
# /etc/rc.d/init.d/httpd.start
```

의 지령을 리용하여 봉사기를 시동시키고

```
# /etc/rc.d/init.d/httpd.stop
```

의 지령을 리용하여 봉사기를 정지시킬수 있다.

주의 자체의 Web 봉사기의 모든 시동과 정지는 기본봉사기프로쎄스가 접속을 기다리는 자식프로쎄스들을 발생시키도록 사용자를 변화시킬수 있게 하기 위하여 뿌리사용자에 의하여 수행되여야 한다.

어쨌든 만일 자기자체의 2진파일들을 설치하거나 원천코드들로부터 새로운 2진파일들을 콤파일하려고 한다면 혹은 구성파일들의 위치를 변화시키려고 한다면 수동으로 httpd지령을 어떻게 써야 하는가를 알 필요가 있을것이다.

보통 httpd은 /usr/sbin에 들어 있는데 두개의 기발을 쓸수 있다.

- -f: httpd.conf 파일에 대한 교체위치를 지적한다.

- -d: 구성파일을 무시하고 봉사기 root를 지적한다.

ServerRoot가 httpd.conf 안에서 지적되기때문에 흔히 -f를 쓰면 충분하다. 실례를 들어 만일 자기의 구성파일들을 /etc/httpd/conf 안에 보관한다면 봉사기를 시동하기 위하여 지령

/usr/sbin/httpd -f /etc/httpd/conf/httpd.conf

를 쓸수 있다.

만일 수동으로 자기의 봉사기를 정지시키고 /etc/rc.d/init.d/httpd를 사용하지 않고 자체로 그것을 시동시키려고 한다면 봉사기에 대한 정확한 프로쎄스 ID(PID)를 알아야 한다. PS 지령을 리용하여 봉사기의 PID를 결정할수 있다.

ps -aux | grep httpd

이 지령은 아래에 보여준것과 같은 프로쎄스들의 목록을 만든다.

```
apache 545  0.1  3.8   1104 572 ? S    17:520:00 [httpd]
apache 546  0.0  3.8   1104572 ? S  17:52  0:00 [httpd]
apache 547  0.2  3.8   1104 572 ? S 17:52 0:00 [httpd]
apache 548  0.1  3.8   1104 572 ? S    17:520:00 [httpd]
apache 549  0.0  3.8   1104 572 ? S 17:52 0:00 [httpd]
apache 550  0.3  3.8   1104 572 ? S    17:520:00 [httpd]
root    544  0.5  4     1104 592 ? S    17:520:00 /usr/sbin/ httpd
```

프로쎄스들이 root에 의하여 차지된 프로쎄스를 제외하고는 모두 apache에 의하여 차지되였다는데 대해 주목한다. 이 프로쎄스는 모든 httpd 프로쎄스들의 부모프로쎄스이며 사용자가 정지시켜야 할 프로쎄스이다. 이때 지령

kill 544

을 써서 정지시킬수 있다.

Apache 사용자와 그룹의 관리

등록부보호에 관한 다음 절에서 보게 되는것처럼 보호의 한가지 기본방법은 사용자이름과 통과암호에 의한것으로서 보호된 등록부에 접근하기 위하여 자기의 사용자이름과 통과암호를 정확하게 제공해야 하는 사용자들의 지적된 목록에로의 접근을 제한하는것이다.

사용자를 만드는것은 간단하며 htpasswd 지령을 리용하여 하는데 그것은 Red Hat Linux와 함께 온 Apache의 기정설치의 경우에 /usr/bin/htpasswd 안에 들어 있다. 이 프로그람은 통과암호파일을 만드는데와 개별적인 사용자들을 만드는데 둘 다 리용된다. 통과암호파일은 사용자이름들과 암호화된 통과암호들을 위한 보관고로서 봉사하는데 그것은 그 후에 Apache에 의하여 접근조종을 위해 리용될수 있다.

지령

htpasswd -c *filename username*

을 리용하여 사용자를 생성할 때 통과암호파일이 처음으로 생성된다.

이 지령을 리용하여 filename 에 의하여 지적된 파일을 만들고 첫 사용자를 그 파일에 추가시킨다. –c 기발은 파일이 만들어 져야 할 필요가 있다는것을 가리킨다. Apache Web 봉사기에 의하여 접근할수 있는 그 어디서나 자기의 통과암호파일을 보관할수 있다. 그것을 보관하기 좋은 장소는 구성파일들이 들어 있는 곳이다. 실례를 들어 지령

 # htpasswd –c /etc/httpd/conf/users user1

은 다른 Apache 구성파일들과 같은 등록부안에 파일 users 를 만들며 사용자 user1 을 만든다.

앞의 지령을 사용할 때 통과암호가 정확하게 들어 가는가를 확인하기 위하여 사용자가 두번 통과암호를 입력할것을 재촉한다. 그다음 통과암호는 암호화되고 /etc/httpd/conf/users 안에 보관된다. 항목은 아래와 같이 된다.

user1:N31mVAxFtiv0

항목의 형식은 사용자이름 다음에 두점, 그다음에 암호화된 통과암호가 있다. 매 사용자항목은 행으로 분리된다.

한편 그것이 체계통과암호파일(/etc/passwd)안에 있는 통과암호로 보이며 Apache 통과암호파일은 같은 방법으로 암호화되는데 htpasswd 에 의해 리용된 암호화처리는 서로 다르며 자기의 Apache 통과암호파일을 만들기 위해 단순히 암호화된 통과암호들을 /etc/passwd 로부터 복사할수 없다.

일단 파일이 만들어 졌으면 추가적인 사용자들을 추가할 때 –c 기발이 없이 간단히 htpasswd 지령을 사용한다.

$ htpasswd /etc/httpd/conf/users user2

이 지령은 사용자 user2 를 추가하는데 역시 통과암호에 대한 두번의 재촉이 일어난다. 결과 통과암호파일은 두개의 항목을 포함할것이다.

user1:N31mVAxFtiv0
user2:WRoTrbH6.3pPK

사용자들을 생성하는외에 보통 그룹들을 창조하려 할것이다. 그룹들은 하나의 이름에 여러명의 사용자들을 함께 련관시키는데 그것은 등록부에 대한 접근조종을 구성할 때 련관된 사용자들을 보다 쉽게 식별할수 있다.

그룹들을 만들려면 vi 혹은 emacs 와 같은 본문편집기를 사용하여 간단하게 파일을 편집하여 groups 파일을 만든다. 아마 /etc/httpd/conf/group 와 같은 식으로 자기의 users 파일과 같은 등록부안에 group 파일을 놓아야 한다. 이 파일은 하나이상의 항목들을 포함하고 있는데 매개는 아래와 같은 형식을 가진 다른 행과 분리되여 있다.

groupname: username1 username2 username3...

실례를 들어 이미 성원들로 된 두명의 사용자들을 가진 authors 라고 부르는 그룹을 만들기 위해서는 아래의 입구점을 리용한다.

authors: user1 user2

접근조종에 의한 등록부보호

《HTML 등록부의 허가》절에서 이미 언급된바와 같이 등록부단위의 접근조종을 구성할수 있다. 보통 이것은 보호하려고 하는 등록부안에 htaccess 라고 부르는 파일을 창조하고 그 파일안에 필요한 구성지령들을 놓는것에 의하여 수행된다.

접근조종의 목적으로 이 파일안에서 리용되는 기본 지령들은 아래와 같다.

- AuthUserFile
- AuthGroupFile
- AuthName
- AuthType
- require
- order
- deny
- allow

AuthUserFile 과 AuthGroupFile

이 지령들은 전번 절에서 만드는 법을 배운 users 와 group 파일들의 위치를 지적하는데 리용한다.

실례에서는 이 지령들이 아래와 같이 나타난다.

```
AuthUserFile   /etc/httpd/conf/users
AuthGroupFile  /etc/httpd/conf/groups
```

이 지령들이 없으면 봉사기가 사용자들과 자기의 통과암호들을 어디서 찾아야 하는지 알지 못하게 되기때문에 이 지령들이 중요하다.

AuthName

AuthName 은 인증령역을 지적하는데 리용된다. 결국 이것은 사용자이름과 통과암호를 요구하는것을 사용자들이 알수 있도록 그들에게 현시되는 재촉문이다. 실례를 들어

```
AuthName Authors only
```

라고 쓰면 사용자이름과 통과암호를 물을 때 사용자에게 Authors Only 재촉문을 현시한다.

Authtype

Authtype 는 렬거된 등록부들우에서 Web 페지들을 호출하는데 리용되는 인증형태를 지적하는데 리용된다. 현재 리용할수 있는 Authtype 선택항목은 오직 basic 로서 이 지령은 apache 의 현행판본들에 거의나 영향을 미치지 않는다.

require

require 지령은 사용자들과 그룹들에 대한 접근을 제한하는데 리용된다. 지령은 통과암호파일안에 있는 모든 사용자들, 특수한 사용자들의 목록, 특수한 그룹들의 목록에 대한 접근을 조종하는데 리용될수 있다.

임의의 존재하고 있는 사용자에 대한 접근을 제한하기 위해서는 지령

require valid-user

을 리용한다.

특수한 사용자들에 대한 접근을 제한하려면

require user *username1 username2 username3…*

형식을 리용한다.

마지막으로 그룹에 대한 접근을 제한하려면 입구점

require group *groupname1 groupname2 groupname3…*

을 리용한다.

order

order 지령은 deny 와 allow 항목들과 함께 리용되여 사용자보다 오히려 주콤퓨터에 기초한 접근을 조종한다. order, deny, allow 를 리용하여 IP 주소 혹은 주콤퓨터이름에 의하여 지적된 주콤퓨터들에 대해서만 접근을 허가하는것이 가능하다.

order 지령은 deny 와 allow 지령들이 적용되여야 하는 순서를 지적한다. 실례를 들어

order allow, deny

는 allow 지령이 처음 적용되여야 하고 만일 의뢰기주콤퓨터가 지령에 의하여 지적된 그러한 사용자들속에 속하지 않는다면 그다음 deny 지령이 적용되여야 한다는것을 말한다.

마찬가지로

order deny, allow

는 순서를 거꾸로 하여 처음에 deny 지령을 적용한다.

deny

deny 지령은 어느 주콤퓨터들의 등록부에 대한 접근을 거절해야 하는가를 지적한다. 가능한 값들은 all, 부분적인 주콤퓨터이름, 부분적 혹은 완전한 IP 주소를 포함한다.

실례를 들어

deny from all

은 모든 주콤퓨터들이 거절된다는것을 의미한다. 이와 비슷하게

deny from .juxta.com

은 juxta.com 령역의 모든 주콤퓨터들의 접근을 거절한다. IP 주소들을 가지는 형식은 아래와 같다.

deny from 194.148.43.195

지령은 지적된 주콤퓨터에 대한 접근을 거절한다.

allow

allow 지령은 deny 지령의 반대로써 어느 주콤퓨터들이 요구한 등록부에 대한 접근을 허가해야 하는가를 지적한다. 그것은 deny 지령에서와 같은 가능한 값들을 가진다.

모든 항목들의 결합

이제는 이 모든것들이 서로 어떻게 결합되는가를 두가지 실례 즉 그룹에 의한 접근허가와 령역이름에 의한 접근허가를 통하여 보기로 하자!

그룹에 의한 접근허가 아래의 실례 .htaccess 파일은 그룹 authors 안에 있는 사용자에 대해서만 지적된 등록부에 대한 접근을 가능하게 한다.

```
AuthName Authors Only
AuthUserFile /etc/httpd/conf/users
AuthGroupFile /etc/httpd/conf/groups
require group authors
```

이 실례는 재촉문에 대하여 AuthName 을 지적하며 통과암호와 그룹파일을 식별하고 그다음 사용자가 등록부를 호출하기 위해서는 그룹 authors 의 한 성원으로 되여야 한다는것을 가리킨다.

령역이름에 대한 접근허가 아래의 실례 .htaccess 파일은 juxta.com 령역안에 있는 주콤퓨터들로부터 Web 싸이트에 접근하는 사용자들에 대해서만 지적된 등록부에 대한 접근을 가능하게 한다.

```
order allow,deny
allow from .juxta.com
deny from all
```

이 실례에서 allow 와 deny 의 순서를 보여 준다. 론리적으로는 다음과 같이 작업하게 된다. 주콤퓨터가 등록부에 접근할 때 처음에 주콤퓨터의 령역을 juxta.com 령역과 비교한다. 만일 주콤퓨터가 지적된 령역안에 속하면 접근은 허가된다. 주콤퓨터가 속하지 않는 경우 deny 지령을 본다. 이 지령은 모든 주콤퓨터들의 접근을 거절해야 한다는것을 나타내므로 요구한 주콤퓨터는 접근이 거절된다.

기록유지

마지막으로 고려해야 할 관리기능은 기록관리이다. Apache 는 두개의 중요한 기록들 즉 접근기록과 오유기록을 산생시킨다. 이와 함께 이것들은 CGI 스크립트들의 오유수정과 자기의 Web 싸이트를 위한 패턴사용법에 대한 구체적인 정보를 가지고 있는 통계적인 접근보고작성을 비롯한 여러가지 기능들에 대한 필요한 정보를 제공한다.

어쨌든 남겨 놓은 이 기록파일은 그자체로 계속 남아 있으므로 크기가 계속 커지면서 상당히 커질수 있다. 그것들이 커질수록 그것들의 가치는 그자체가 포함하고 있는 자료량들을 처리하는것이 더 힘들어 지기때문에 감소된다.

이러한 리유로 해서 자기의 기록들을 순환시키기 위한 일정작성을 만들어 놓는것이 좋다. 기록들을 순환시킨다는것은 존재하고 있는 판본들을 보관고에 보관하고 새로운 빈 기록을 가지고 다시 시작한다는것을 의미한다. 얼마나 자주 기록을 받으려는가에 따라서 기록들을 순환시키기 위한 리상적인 빈도는 달로부터 주, 일에 이르기까지 임의로 될수 있다.

Red Hat Linux 7.1 은 /var/log/httpd 등록부에 Apache 등록들을 병합시킨다. 그것들은 /etc/cron.daily/logrotate 스크립트안에 정의된 일별과제로써 /var/log 안에 있는 다른 등록들과 함께 순환된다.

대신에 Apache 의 내리적재된 판본을 설치하였다면 자기의 등록들을 자동적으로 순

환시키는 rotatelogs 편의프로그람을 리용할수 있다. 기정으로 Red Hat Linux 7.1 은 /usr/sbin/rotatelogs 에 이것을 설치한다. 프로그람을 리용하기 위하여서는 httpd.conf 에 아래의 항목을 추가해야 한다.

TransferLog "|/usr/sbin/rotatelogs /some/location/file time"

항목 /some/lcoation/file 은 순환된 기록들에 대한 초보적인 파일이름을 제공한다. 기록이 시작된 체계시간을 표시하는 수자는 파일이름에 부가된다. time 파라메터는 얼마나 자주 기록들을 순환하려고 하는가를 초단위로 지적한다.

Web 싸이트만들기

자기의 Web 싸이트를 구성하고 원활하게 실행하게 되였으므로 Web 에 대한 내용을 어떻게 전개하는가를 보여 주는 작은 실례 Web 싸이트를 만들자

만들려고 하는 싸이트는 On the Web Publishers 라고 부르게 될 작은 출판회사에 대한 정보를 가지게 된다. 이 싸이트는 새로운 제목들의 목록, 출판사에 대한 일반적인 정보, 련계되는 양식, 출판사와 계약을 맺는 저자들이 배타적으로 리용하도록 통과암호로 보호를 한 부분을 제공한다.

모두 협력하여 HTML 과 영상들을 포함하며 CGI 프로그람작성을 리용하면서 싸이트의 통과암호가 보호된 부분에 대한 접근조종을 실현하는 싸이트를 만들어 보자. 우선 그림 32-2 에서처럼 자기 싸이트의 구조를 작성하여야 한다.

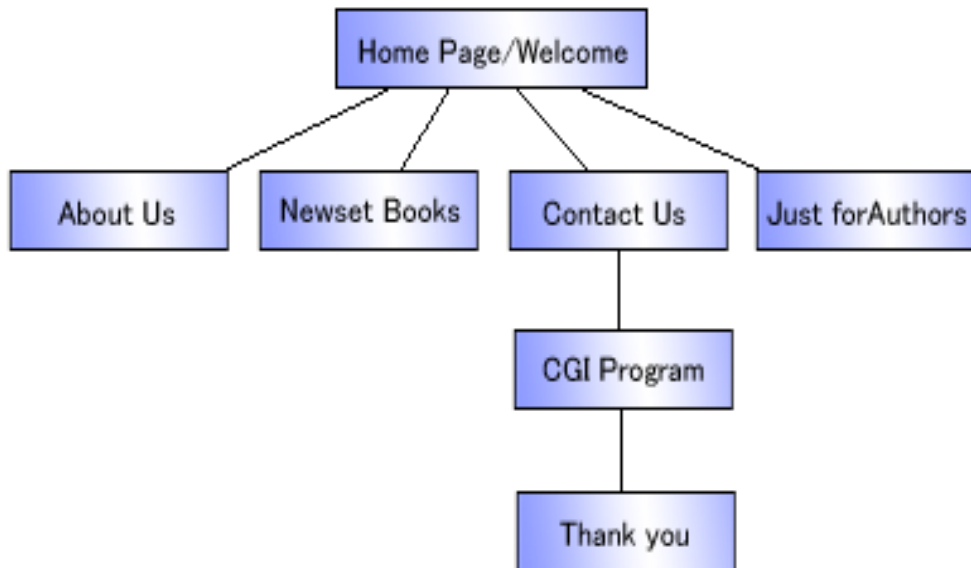



그림 32-2. Web 싸이트의 구조

이것을 한 다음 이 구조를 실현하는데 필요한 파일들을 만드는것이 필요하다. 이 장의 앞선 절들에서 리용된 구성을 가정하면 HTML 파일들에 대한 뿌리문서나무는 /var/www/html 에 있고 CGI 등록부는 /var/www/cgi-bin 에 있다. 그래서 우리의 싸이트에 대한 등록부와 파일나무는 아래 파일들을 포함해야 할것이다.

```
/var/www/html/index.html
/var/www/html/about/index.html
/var/www/html/books/index.html
/var/www/html/contact/index.html
/var/www/html/authors/index.html
/var/www/cgi-bin/formmail
```

후에 간단히 론의하게 될 마지막파일을 제외하고 이 모든 파일들은 HTML 파일이다. HTML 파일들에 의하여 리용되는 임의의 지원영상들은 HTML 파일들과 같은 등록부안에 놓일수 있지만 그것들을 완전히 보존하기 위하여 대부분의 Web 관리기들은 모든 자기들의 영상들을 다른 등록부에 놓는것을 오히려 더 좋아한다. 이 경우에 그 등록부는

```
/var/www/html/images/
```

와 같이 될수 있다.

이 장에서는 HTML 을 배우자는것은 아니므로 그것들이 어떻게 작성되는가 하는 실례들로 두개의 HTML 파일을 보기로 하자. 기본홈페지를 /var/www/html/index.html 로부터 시작하자. 실례에서 이 파일에 대한 원천코드는 아래와 같다.

```
<HTML>

    <HEAD>
    <TITLE>On the Web Publishers</TITLE>
    </HEAD>
    <BODY BGCOLOR=lightcyan TEXT=midmghtblue>
    <DIV ALIGN=CENTER>
        <A HREF="/"><IMG SRC"/images/logo.gif" BORDER=O></A>
        <TABLE BORDER=0 CELLPADDING=5 CELLSPACING=5
            BGCOLOR=1ightpink>
        <TR>
            <TD ALIGN=CENTER><A HREF="about">ABOUT
                US</A></TD>
            <TD ALIGN=CENTER><A HREF="books">OUR
                BOOKS</A></TD>
            <TD ALIGN=CENTER><A HREF="contact">CONTACT
                US</A></TD>
            <TD ALIGN<ENTERXA HREF="authors">]UST FOR
                AUTHORS</A></TD>
        </TR>
        </TABLE>
     </DIV>
```

```
        <FONT SIZE=5>W</F>e1come to <STRONG>On the Web
        Pub1ishers</STRONG>. We offer the finest in
        on-line electronic books at reasonable prices.
        Check out what we have to offer ...

        <UL>
        <LI><A HREF="about">Learn about what we do</A>
        <LI><A HREF="books">See what books we offer</A>
        <LI><A HREF="contact">Contact us</A>
        </UL>

        </BODY>
    </HTML>
```

이 파일은 그림 32-3 에서와 같은 결과들을 산생한다.

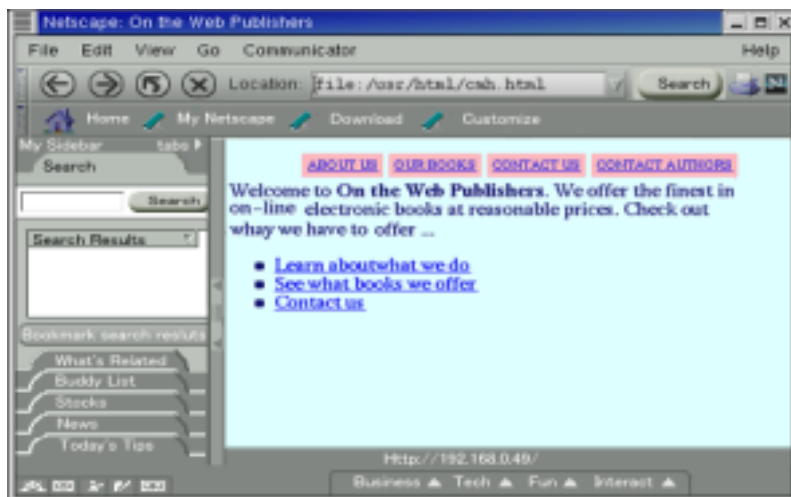



그림 32-3. 기본홈페지

다른 페지들은 /var/www/html/contact/index.html 에서 접촉양식을 제외하고는 원천코드가 비슷하다. 이 페지는 다음의 원천코드에 의하여 산생되는데 그림 32-4 와 같이 보이게 된다.

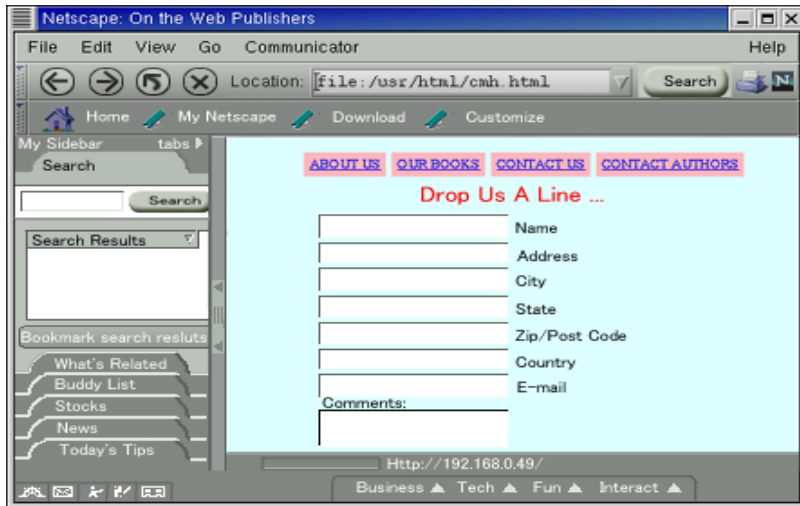



그림 32-4 .접촉양식

```
<HTML>

    <HEAD>
        <TITLE>On the Web Publishers</TITLE>
        </HEAD>

        <BODY BGCOLOR=lightcyan TEXT=midnightblue>
        <DIV ALIGN=CENTER>
            <A HREF="/"><IMG SRC= "/images/logo.gif" BORDER=O></A>
            <TABLE BORDER=0 CELLPADDING=5 CELLSPACING=5
                BGCOLOR=lightpink>
            <TR>
                <TD ALIGN=CENTERXA HREF="/about">ABOUT
                    US</A></TD>
                <TD ALIGN=CENTER><A HREF="/books">OUR
                    BOOKS</A></TD>
                <TD ALIGN=CENTER BGCOLOR=yellow>CONTACT
                    US</TD>
                <TD ALIGN=CENTERXA HREF="/authors">]UST
                  FOR
                    AUTHORS</A></TD>
            </TR>
            </TABLE>
```

```
                <Hl>Drop Us A Line .. .</H1>
      </DIV>
      <TABLE ALIGN=CENTER><TR><TD>
      <FORM METHOD=POST ACTION="/cgi-bi n/formmail">
      <INPUT TYPE=TEXT WIDTH=30 NAME=name> Name<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=address> Address<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=city> City<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=state> State<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=zip> Zip/Post Code<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=country> Country<BR>
      <INPUT TYPE=TEXT WIDTH=30 NAME=emai1> E-mail <BR>
      Comments:<BR>
      <TEXTAREA ROWS=10 COLS=30 NAME=comments
          WRAP=HARD></TEXTAREA><BR>
      <INPUT TYPE=SUBMIT VALUE="Send Comments'>
      </FORM>
      </TD></TR></TABLE>
      </BODY>

   </HTML>
```

이 페지에서 특수한것은 그것이 양식과 양식안에 있는 자료를 처리하는 CGI 프로그람에 대한 참조를 포함하는것이다. 이 경우 리용하고 있는 프로그람은 formmail 이라고 부르는데 그것은 Perl 로 작성된 자유롭게 리용할수 있는 CGI 스크립트로서 그것은 양식의 내용들을 장악해서 그것들을 이미 정의된 우편주소의 우편으로 보낸다. 이 방식에서 양식으로부터의 접촉정보는 책방의 기본전자우편주소에 전자우편으로 갈수 있다.

Formmail 은 Matthew M.Wright 에 의하여 작성되였으며 http://www.worldwidemart.com/scripts/formmail .shtml 에서 리용할수 있다. 비록 이 장에서는 Perl 와 CGI 프로그람작성을 론의하지 않지만 품들이지 않고 기초적인 CGI 프로그람들을 만드는것이 단순하다는것을 보여 주기 위해 원천코드를 여기에 제시한다.

```
   #!/usr/bin/per1
   ##############################################################################
   # FormMail          Version 1.6                       #
   # Copyright 1995-1997 Matt Wright mattw@wor1dwidemart.com      #
   # Created 06/09/95       Last Modified 05/02/97            #
   #Matt'sScriptArchive,Inc.: http://www.wor1dwidemart.com/scripts/#
   ##############################################################################
   #COPYRIGHT NOTICE                                 #
   #Copyright 1995-1997 Matthew M. Wright All Rights Reserved.     #
```

```
#                                               #
# FormMail may be used and modified free of charge by anyone so long #
as this copyright notice and the comments above remain intact. By using #
this                                                                  #
# code you agree to indemnify Matthew M. Wright from any liability that#
# might arise from its use                                            #
#                                            #
# Selling the code for this program without prior written consent is    #
# expressly forbidden. In other words, please ask first before you tryand#
# make money off of my program                            #
#                                            #
# Obtain permission before redistributing this software over the Internet #
or in any other medium. In all cases copyright and header must remain # intact
                                                          #
############################################################################
# Define Variables                                    #
#  Detailed Information Found In README File                          #
# $mailprog defines the location of your sendmail program on your unix#
# system                                                              #
$mai1prog = '/usr/lib/sendmail';
# @referers allows forms to be located omy on servers wmcn are defined#
# in this field. This security fix from the last version which allowed    #
# anyone on any server to use your FormMail script on their web site.#
@referers = ('1inux.juxta.com');
# Done                                 #
############################################################################
#   Check   Referring   URL
&check_ur1;
# Retrieve Date
&get_date;
# Parse Form Contents
&parse_form;
# Check Required Fields
&check_required;
# Return HTML Page or Redirect User
&return_htm1;
# Send E-Mail
&send_mai1;
sub check_ur1 {
```

```
# Localize the check_referer flag which determines if user is valid.#
local($check_referer) = 0;
# If a referring URL was specified, for each valid referer, make sure#
# that a valid referring URL was passed to FormMail. #
if ($ENV{'HTTP_REFERER'}) {
    foreach $referer (@referers) {
        if($ENV{'HTTP_REFERER'}=~m1https?//([^/]*)$referer1i){
            $check_referer = 1;
             last;
         }
        }
    }
    else {
        $check_referer = 1;
}
# If the HTTP_REFERER was invalid, send back an error. #
if ($check_referer != 1) { &error('bad_referer') }
}
sub get_date {
    # Define arrays for the day of the week and month of the year. #
    ®days = ('Sunday','Monday','Tuesday','Wednesday',
            'Thursday','Friday','Saturday');
    ©months=('January','February','March','April','May','June',
            'July','August','September','October','November',
            'December');
    # Get the current time and format the hour, minutes and seconds.
    # Add 1900 to the year to get the full 4 digit year.#
    ($sec,$min,$hour,$mday,$mon,$year,$wday)=(1ocaltime(time))[0,1
       ,2,3,4,5,6];
    $time = spr-infr("%02d:%02d:%02d",$hour,$min,$sec);
    $year += 1900;
    #Format the date.  #
    $date = "$days[$wday], $months[$mon] $mday, $year at $time";
}
sub parse_form {
    #Define the configuration associative array. #
```

```
%Config = ('recipient','',          'subject','',
            'emai1','',                       'realname','',
            'redirect','',            'bgcolor','',
            'background','',          '1ink_co1or','',
            'v1ink_co1or' , ' ' ,            'text_co1or' , ' ' ,
            'a1ink_co1or1 , ' ' ,    'title','',
            'sort','',                        'print_config', ' ' ,
            'required','',                    'env_report','' ,
            'return_1ink_tit1e','',           'return_1ink_ur1','',
            'print_b1ank_fields','',  'missing_fields_redirect','');
# Determine the form's REQUEST_METHOD (GET or POST) and split t
# he form fields up into their name-value pairs. If the REQUEST_METH
# OD was                                                          #
# not GET or POST, send an error.                     #
if ($ENV{'REQUEST_METHOD'} eq 'GET') {
    # Split the name-value pairs
    ©pairs = sp1it(/&/, $ENV{'QUERY_STRING'});
}
elseif ($ENV{'REQUEST_METHOD'} eq 'POST') {
    # Get the input
    read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});

    # Split the name-value pairs
    ©pairs = sp1it(/&/, $buffer);
}
else {
    &error('request_method');
}
# For each name-value pair:                             #
foreach $pair (©pairs) {
    # Split the pair up into individual variables.      #
    local($name, $value) = sp1it(/=/, $pair);
    # Decode the form encoding on the name and value variables.#
    $name =~ tr/+/ /;
    $name=~s/%([a-fA-FO-9][a-fA-FO-9])/pack("C",hex($l))/eg;
    $value =~ tr/+/ /;
    $value=~s/%([a-fA-FO-9][a-fA-FO-9])/pack("C",hex($l))/eg;
            # If they try to include server side includes, erase them,
```

```
          so # they aren't a security risk if the html gets returned.
          Another
# security hole plugged up.
$value =~ s/<!-(.|\ n)*->//g;
            # If the field name has been specified in the %Config array,
          # it will return a 1 for defined($Config{$name})) and we shou
          # ld associate                                              #
            #this value with the appropriate configuration variable.If
          this
          #
            #is not a configuration form field,put it into the associative#
            # array %Form, appending the value with a ', 'if there is
          alrea # dy a value present. We also save the order of the form
          fields # in the                                             #
            # @Fie1d_Order array so we can use this order for the generic
          # sort,                                                 #
if (defined($Conf1g{$name})) {
             $Config{$name} = $value;
          }
          else {
               if ($Form{$name} && $value) {
                  $Form{$name} = "$Form{$name}, $value";
               }
               elsif ($value) {
                  push(@Fie1d_Order,$name);
                  $Form{$name} = $value;
               }
          }
     }
  # The next six lines remove any extra spaces or new lines from the#
  # configuration variables, which may have been caused if your editor#
  # wraps lines after a certain length or if you used spaces between field#
  #names or environment variables.    #
  $Config{'required'} =" s/\ s+|\ n)?,( \ s+|\ n)?/,/g;
  $Config{'required'} =~ s/(\ s+)?\ n+(\ s+)?//g;
  $Config{'env_report'} =~ s/(\ s+|\ n)?,( \ s+|\ n)?/,/g;
  $Config{'env_report'} =~ s/(\ s+)?\ n+(\ s+)?//g;
  $Config{'print_config'} =~ s/(\ s+|\ n)?,( \ s+|\ n)?/,/g;
```

```
        $Config{'print_config'} =~ s/(\ s+)?\ n+(\ s+)?//g;
        # Split the configuration variables into individual field names.  #
        @Required = sp1it(/,/,$Config{'required'});
        @Env_Report = sp1it(/,/,$Config{'env_report'});
        @Print_Config = sp1it(/,/,$Config{'print_config'});
    }
   sub check_required {
        # Localize the variables used in this subroutine.  #
        1oca1($require, @error);
        if (!$Config{' recipient'}) {
            if (!defined(%Form)) { &error('bad_referer') }
            else          { &error('no_rec1pient') }
        }
        # For each require field defined in the form:              #
        foreach $require (@Required) {
            # If the required field is the email field, the syntax of the email#
            #address if checked to make sure it passes a valid syntax.      #
            if ($require eq 'email' && !&check_email($Conf1g{$require})) {
                push(@error,$require);
            }
            # Otherwise, if the required field is a configuration field and it  #
            # has no value or has been filled in with a space, send an error.#
            elsif (defined($Config{$require})) {
                if (!$Config{$require}) {
                push(@error,$require);
            }
        }
        # If it is a regular form field which has not been filled in or      #
        # filled in with a space, flag it as an error field.            #
        elsif (!$Form{$require}) {
            push(@error,$require);
         }
      }
      # If any error fields have been found, send error message to the user.#
      if (©error) { &error('missing_fields', ©error) }
      }
```

```
sub return_html {
    # Local variables used in this subroutine initialized.          #
    local($key,$sort_order,$sorted_field);
    # If redirect option is used, print the redirectional location header.#
    if ($Config{'redirect'}) {
       print "Location: $Config{'redirect'}\ n\ n";
    }
    # Otherwise, begin printing the response page.            #
    else {
        # Print HTTP header and opening HTML tags          #
        print "Content-type: text/html\ n\ n";
        print "<html>\ n <head>\ n";
        # Print out title of page                      #
        if ($Config{'title'}) { print " <title>$Config{'title'}</title>\
        n" }
        else          { print " <title>Thank You</title>\ n"  }
        print " </head>\ n <body";
        # Get Body Tag Attributes                     #
        &body_attriibutes;
        # Close Body Tag                          #
        print ">\ n <center>\ n";
        # Print custom or generic title.                  #
        if ($Config{'title'}) { print "  <hl>$Config{'title'}</hl>\ n" }
        else ( print " <hl>Thank You For Filling Out This Form</hl>
        \ n" )
        print "</center>\ n";
        print "Below is what you submitted to $Config{'recipient'} on ";
        print "$date<p><hr size=l width=75\ %><p>\ n";
        # Sort alphabetically if specified:               #
        if ($Config{'sort'} eq 'alphabetic') {
           foreach $field (sort keys %Form) {
           # If the field has a value or the print blank fields option#
           # is turned on, print out the form field and value.      #
           if ($Config{'print_blank_fields'} || $Form{$field}) {
              print "<b>$field:</b> $Form{$field}<p>\ n";
```

```
        }
      }
 }
# If a sort order is specified, sort the form fields based on that.  #
elsif ($Config{'sort'} =~ /^order:.*,.*/) {
   # Set the temporary $sort_order variable to the sorting order,  #
   # remove extraneous line breaks and spaces, remove the order:#
   # directive and split the sort fields into an array.        #
   $sort_order = $Config{'sort'};
    $sort_order =~ s/(\ s+|\ n)?,( \ s+|\ n)?/,/g;
    $sort_order =~ s/(\ s+)?\ n+(\ s+)?//g;
    $sort_order =~ s/order://;
    @sorted_fields = split(/,/, $sort_order);
   # For each sorted field, if it has a value or the print blank     #
   # fields option is turned on print the form field and value.     #
   foreach $sorted_field (@sorted_fields) {
       if ($Config{'print_blank_fields'} || $Form{$sorted_field}{
          print"<b>$sorted_field;</b>$Form{$sorted_field}<p>\ n";
       }
   }
 }
 # Otherwise, default to the order in which the fields were sent.#
 else {
      # For each form field, if it has a value or the print blank    #
      # fields option is turned on print the form field and value.  #
      foreach $field (@Field_Order) {
           if ($Config{'print_blank_fields'} || $Form{$field}) {
              print "<b>$field:</b> $Form{$field}<p>\ n" ;
           }
        }
 }
 print "<p><hr size=l width=75%><p>\ n";
 # Check for a Return Link and print one if found.  #
 if ($Config{'return_link_url'} && $Config('return_link_title')) {
    print "<ul>\ n";
    print"<li><a  href= \  "$Config{'return_link_url1} \  ">$Config
```

```
        {'return_
          link_title'}</a>\ n";
        print "</ul>\ n";
    }
    # Print the page footer. #
    print «"(END HTML FOOTER)";
    <hr size=l width=75%<p>
    <center><font size=-1><a href="http://www.worldwidemart.com/
    scr1pts/
      formmail.shtml">FormMail</a> VI.6 &copy; 1995 -1997 Matt
    Wr1ght<br>
A Free Product of <a href="http://www.worldwidemart.com/scr1pts/">
Matt's
  Script Archive, Inc.</a></font></center>
     </body>
    </html>
(END HTML FOOTER)
    }
}
sub send_mail {
    # Localize variables used in this subroutine.          #
  local($print_config,$key,$sort_order,$sorted_field,$env_report);
    # Open The Mail Program
    open(MAIL,"|$mailprog -t");
    print MAIL "To: $Config('recipient')\ n";
    print MAIL "From: $Config{'email'} ($Config{'realname'})¥n";
    # Check for Message Subject
    #if ($Config{'subject'}) { print MAIL "Subject: $Config{'subject'}
    \ n\ n" }
    else         { print MAIL "Subject: WWW Form Submission\ n\
    n" }
    print MAIL "Below is the result of your feedback form. It was
   submitted
     by\ n";
   print MAIL "$Config('realname') ($Config{'email'}) on $date\ n";
```

```
 print MAIL "-" x 75 . "\ n\ n";
 if (@Print_Config) {
     foreach $print_config (@Print_Config) {
         if ($Config{$print_config}) {
            print MAIL "$print_conf1g: $Config($print_config) \ n\
            n";
         }
     }
 }
# Sort alphabetically if specified:                    #
if ($Config{'sort'} eq 'alphabetic') {
    foreach $field (sort keys %Form) {
        # If the field has a value or the print blank fields option    #
        # is turned on, print out the form field and value.        #
        if ($Config{'print_b1ank_fields'} || $Form{$fie1d} ||
            $Form{$fie1d} eq '0') {
            print MAIL "$fie1d: $Form{$fie1d}\ n\ n";
      }
 }
}
# If a sort order is specified, sort the form fields based on that.    #
elsif ($Config{'sort'} =~ /^order: .*,.*/) {
    # Remove extraneous line breaks and spaces, remove the order:#
    # directive and split the sort fields into an array.            #
    #Config{'sort'} =~ s/(\  s+|\  n)?,( \  s+|\  n)?/,/g:
    $Config{'sort'} =~ s/(\ s+)?\ n+(\ s+)?//g;
    $Config{'sort'} =~s/order://;
    @sorted_fie1ds = split(/,/, $Config{'sort'});
    # For each sorted field, if it has a value or the print blank    #
    # fields option is turned on print the form field and value.    #
    foreach $sorted_fie1d (@sorted_fie1ds) {
        if ($Config{'print_b1ank_fields'} || $Form($sorted_fie1d) ||
           $Formi{$sorted_fie1d} eq '0') {
           print MAIL "$sorted_fie1d: $Form{$sorted_fie1dl}\ n\ n';
        }
    }
}
```

```
# Otherwise, default to the order in which the fields were sent.      #
 else {
     # For each form field, if it has a value or the print blank      #
     # fields option is turned on print the form field and value.     #
     foreach $field (@Fie1d_Order) {
         if ($Config{'print_blank.fields'} || $Form{$field} ||
            $Form{$field} eq '0') {
            print MAIL "$field: $Form{$fie1d}\ n\ n";
         }
     }
 }
print MAIL "-" x 75 . "\ n\ n";
# Send any specified Environment Variables to recipient.        #
foreach $env_report (@Env_Report) {
    if ($ENV{$env_report}) {
        print MAIL "$env_report: $ENV{$env_report} \ n";
    }
}
   close (MAIL);
}
sub check_email1 {
# Initialize local email variable with input to subroutine.      #
$email = $_[0];
# If the e-mail address contains:                         #
if ($email =~ /(@.*@)|(\ . \ .)| (@\ .)|( \ .@)| (^\ )/ | |
   # the e-mail address contains an invalid syntax. Or, if the    #
   # syntax does not match the following regular expression pattern#
   # it fails basic syntax verification.             #
   $email !~ /^.+\ @(\ [?][a-zA-ZO-9 \ -\ .]+ \ .([a-zA-Z] {2,3} |
   [0-9] {1,3})(\ )?)
     $/) {
   # Basic syntax requires: one or more characters before the @ sign,#
   # followed by an optional '[', then any number of letters, num
   # bers,                                                           #
   # dashes or periods (valid domain/IP characters) ending in a period#
   # and then 2 or 3 letters (for domain suffixes) or 1 to 3 numbers#
```

```
      # (for IP addresses). An ending bracket is also allowed as it is#
      # valid syntax to have an email address like: user@[255.255.255]   #
      # Return a false value, since the e-mail address did not pass valid#
      # syntax.                                  #
      return 0;
   }
   else {
       # Return a true value, e-mail verification passed.            #
       return 1;
   }
}
sub body_attributes {
     # Check for Background Color
     if ($Config{'bgcolor'}) { print " bgcolor=\ "$Config{'bgcolor'} \
     "" }
     # Check for Background Image
     #if  ($Config{'background'})  {  print  "  >-  background= \
     "$Config{ 'background' }\
       ""}
     # Check for Link Color
     #if ($Config{'link_color'}) { print " link=\ "$Config{'link_color'}
     \ "" }
     # Check for Visited Link Color
     if ($Config{'vlink_color'}) { print " vlink=\ "$Conf1g{'vlink_col
     or' }\ "" }
     # Check for Active Link Color
     if ($Config['alink_color']) { print " alink=\ "$Config{'alink_colo
     r'}\ "" }
     # Check for Body Text Color
     if ($Config{'text_color'}) { print " text=\ "$Config{'text_color'}
     \ "" }
}
sub error {
     # Localize variables and assign subroutine input.           #
     local($error,@error_fields) = @_;
     local($host,$missing_field,$missing_field_list);
     if ($error eq 'bad_referer') {
        if ($ENV{'HTTP_REFERER'} =~ m|^https?://([ \ w \ .]+) |i) {
            $host = $1;
            print <<"(END ERROR HTML)";
Content-type: text/html
```

```
<html>
<head>
<title>Bad Referrer - Access Denied</title>
</head>
<body bgcolor=#FFFFFF text=#000000>
<center>
 <table border=0 width=600 bgcolor=#9C9C9C>
  <tr><th><font size=+2>Bad Referrer - Access Denied</font></th></tr>
</table>
<table border=0 width=600 bgcolor=#CFCFCF>
<tr><td>The form attempting to use
<a href="http://www.worldwidemart.com/scripts/formmail.shtml">
FormMail</a>
resides at <tt>$ENV{'HTTP_REFERER'}</tt>, which is not allowed to
access this cgi script.<p>
If you are attempting to configure FormMail to run with this form, you
  need
to add the following to \@referers, explained in detail in the README
  file.<p>
Add <tt>'$host'</tt> to your <tt><b>\@referers</b></tt> array.<hr size=l>
<center><font size=-l>
<a href="http://www.worldwidemart.com/scripts/formrnail.shtmt">Fo
rmMail
  </a> V1.6 &copy; 1995 - 1997 Matt Wright<br>
A Free Product of <a href=http://www.worldwidemart.com/scripts/"
>Matt's
  Script Archive, Inc.</a>
</font></center>
</td><tr>
</table>
</center>
</body>
</html>
(END ERROR HTML)
```

```
        }
      else {
            print <<"(END ERROR HTML)";
Content-type: text/html
<html>
<head>
<title>FormMail v1.6</title>
</head>
<body bgcolor=#FFFFFF text=#000000>
<center>
<table border=0 width=600 bgcolor=#9C9C9C>
<tr><th><font size=+2>FormMail</font></th></tr>
</table>
<table border=0 width=600 bgcolor=#CFCFCF>
<tr><th><tt><font size=+1>Copyright 1995 - 1997 Matt Wright<br>
   Version 1.6 - Released May 02, 1997<br>
   A Free Product of <a href='http://www.worldwidemart.com/
   scripts/">
     Matt's Script Archive,
   Inc. </a></font></tt></th></tr>
</table>
</center>
</body>
</html>
(END ERROR HTML)
            }
      }
      elsif ($error eq 'request_method') {
            print <<"(END ERROR HTML)";
Content-type: text/html

<html>
<head>
<title>Error: Request Method</title>
</head>
```

```
<body bgco1or=#FFFFFF text=#000000>
<center>
<tab1e border=0 width=600 bgco1or=#9C9C9C>
<tr><th><font size=+2>Error: Request Method</font></th></tr>
</tab1e>
<tab1e border=0 width=600 bgco1or=#CFCFCF>
<tr><td>The Request Method of the Form you submitted did not match
either <tt>GET</tt> or <tt>POST</tt>. Please check the form and make
  sure the
<tt>method=</tt> statement is in upper case and matches <tt>GET</tt> or
  <tt>POST</tt>.<p>
<center><font size=-l>
    <a   href="http://www.wor1dwidemart.com/scripts/formmai1.shtml
">ForMai1
  </a> V1.6 &copy; 1995 - 1997 Matt Wnght<br>
A Free Product of <a href="http://www.wor1dwidemart.com/scripts/
">Matt's
  Script Archive, Inc.</a>
</font></center>
</td></tr>
</tab1e>
</center>
</body>
</htm1>
(END ERROR HTML)
     }
     elsif ($error eq 'no_recipient') {
             print <<"(END ERROR HTML)";
Content-type: text/html
<htm1>
<head>
<tit1e>Error: No Recipient</tit1e>
</head>
<body bgco1or=#FFFFFF text=#000000>
```

```
<center>
<table border=0 width=600 bgcolor=#9C9C9C>
<tr><th><font size=+2>Error: No Recipient</font></th></tr>
</table>
<table border=0 width=600 bgcolor=#CFCFCF>
<tr><td>No Recipient was specified in the data sent to FormMail. Please
make sure you have fined in the 'recipient' form field with an e-mail
address. More information on filling in recipient form fields can be
found in the README file.<hr size=l>
<center><font size=-l>
<a href="http://www.worldwidemart.com/scripts/formmail.shtml"> For
mMail
  </a> V1.6 &copy; 1995 - 1997 Matt Wright<br>
A Free Product of <a href="http://www.worldwidemart.com/scripts/">
Matt's
  Script Archive, Inc.</a>
</font></center>
</td></tr>
</table>
</center>
</body>
</html>
(END ERROR HTML)
      }
      elsif ($error eq 'missing_fields') {
         if ($Config{'missing_fields_redirect'}) {
              print "Location: $Config('missing_fields_redirect' ) \ n\
               n";
         }
         else {
            foreach $missing_field (@error_fields) {
                $missing_field_list .= "   <li>$missing_field\ n";
            }
                print <<"(END ERROR HTML)";
 Content-type: text/html
<html>
```

```
<head>
  <title>Error: Blank Fields</title>
</head>
<center>
<table border=0 width=600 bgcolor=#9C9C9C>
  <tr><th><font size=+2>Error: Blank Fields</font></th></tr>
</table>
<table border-0 width=600 bgcolor=#CFCFCF>
<tr><td>The following fields were left blank in your submission form:<p>
  <ul>
$missing_field_list
  </ul><br>
 These fields must be filled in before you can successfully submit the
    form.<p>
  Please use your browser's back button to return to the form and try
    again.<hr size=l>
  <center><font size=-l>
  <a   href="   http://www.worldwidemart.com/scripts/formmail.shtml
  ">FormMail
    </a> V1.6 &copy; 1995 - 1997 Matt Wright<br>
  A Free Product of <a href=http://www.worldwidemart.com/scripts/">
  Matt'S
    Script Archive, Inc.</a>
  </font></center>
 </td></tr>
</table>
</center>
</body>
</html>
(END ERROR HTML)
         }
      }
    exit;
}
```

주목해야 할것은 이 프로그람이 양식안에 있는 접촉정보가 처리되였다는것을 가리키거나 오유가 있으면 오유를 가리키는 HTML 페지를 사용자에게 돌려 주도록 설계되였다는것이다. 이것은 모든 CGI 프로그람들의 중요한 구성요소인바 그것들은 열람기에 유효한 자료(HTML 혹은 영상파일의 내용들과 같은)를 되돌려 주어야 하거나 아니면 그것들은 열람기가 유효한 URL에로 방향바꾸기해야 한다.

formmail프로그람은 전자우편들의 제목행을 설정하고 다른 Web싸이트우의 사용자에 의하여 스크립트가 람용되지 않는다는것을 확인할수 있는 능력 등 많은 다

른 특징들을 제공한다. 완전히 자세한 내용들은 프로그람의 Web 싸이트에서 볼수 있다.

Web 싸이트의 마지막으로 중요한 부분은 등록부 /var/www/html/authors/에 대하여 오직 인증된 사용자들만이 이 등록부안에 있는 파일들에 접근할수 있도록 접근제한들을 만드는것이다. 이렇게 하는것은 세 단계의 처리를 요구한다.

첫째로, 《접근조종에 의한 등록부보호》의 마지막 소절에서 한것처럼 그안에 필요한 사용자들을 가지는 사용자들의 파일을 만들 필요가 있다. 등록부에 대한 접근을 주려고 하는 모든 사용자들에 대한 접근제한들을 만든다.

둘째로, 등록부에 대한 접근을 더 쉽게 관리하는 일을 하기 위하여 그룹(authors 라고 부르자)을 만들려고 한다. 아래와 같이 자기의 Web 봉사기의 group 파일에 그룹에 대한 항목을 추가하는것에 의하여 이 그룹을 만든다.

authors: *author1 author2 author3*

이 항목은 등록부에 대한 접근을 가지고 있는 세명의 사용자가 있다는 것을 말해 준다.

마지막으로 /var/www/html/authors/라고 부르는 .htaccess 파일을 만들 필요가 있다. (이 지령들속에 있는 "Auth"는 인증에 기인되는것이지 다른 authors 그룹에 대한 것이 아니다.)

```
AuthName Just For Authors
AuthType Basic
AuthUserFile /etc/httpd/conf/users
AuthGroupFile /etc/httpd/conf/groups
require group authors
```

이 파일의 내용은 대단히 간단하다. AuthName 은 사용자에게 보내여지는 재촉문을 지적한다. AuthUserFile 은 봉사기에게 어디서 적당한 사용자이름들과 통과암호들의 목록을 찾을수 있는가를 알려 주며 마찬가지로 AuthGroupFile 은 봉사기에게 어디에 적당한 그룹들의 목록이 있는가를 알려 준다. 마지막으로 require 지령은 오직 그룹 authors 의 성원들만이 등록부에 대한 접근을 해야 한다는것을 가리킨다.

이제 사용자가 대화조종에서 처음으로 요구한 등록부에 접근하려고 할 때 그들은 Netscape 에 의하여 만들어 진 그림 32-5 에 있는것과 같은 자기의 열람기에 의하여 현시된 인증대화칸을 볼것이다.

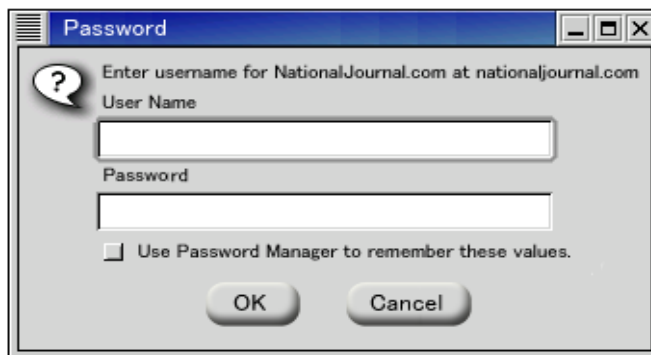



그림 32-5. Netscape 의 인증대화칸

기정으로 사용자가 인증을 적당하게 하지 못하여 실패했다면 그들은 인증이 필요되

는 상태를 표시하는 일반적인 오유페지를 받게 된다. 고장을 가리키는 전용화된 페지를 제공하기 위해서는 그 파일에 아래의 항목을 추가함으로써 전용화된 오유통보문을 지적하는 httpd.conf 파일을 편집할수 있다.

ErrorDocument 401 /error.html

이 항목은 오유 401(인증오유)이 일어 났을 때 봉사기가 기정통보문대신에 지적된 URL 을 돌려 보내야 한다는것을 나타낸다. 현재 사용자가 인증에서 실패했다면 그들에게 자기의 봉사기에 대하여 서명할것을 알려 주는 자기의 전용화된 통보문을 받을것이다.

▶미리 알아두기

이 장은 Linux 조작체계가 망봉사기로서, 이 경우에는 Web 봉사기로서의 유연한 도구를 어떻게 제공하는가를 보여 주었다.

제 33 장 《우편봉사기로서의 Linux(Sendmail 능력)》은 Linux 가 담당할수 있는 다른 중요한 역할 즉 우편봉사기로서의 역할을 시험한다. 우편봉사기는 망에서 들어 오고 나가는 모든 우편들을 처리하기, 통보문들이 내부사용자들을 위해 미리 정해 지도록 결정하기, 통보문들을 사용자들의 우편통들에 안내하기, 배달을 위하여 인터네트우의 외부망범위에서 통보문들을 안내하기를 조종한다.

특히 유력한 Linux의 우편봉사기인 Sendmail 프로그람을 리용하여 우편봉사기를 설치하는것은 Unix 환경에서 힘든 과제일수 있다. 그러나 그 일을 관리할수 있는 대부분의 망들에서 그것을 일단 수행하면 Linux 우편봉사기는 거의 날마다 주고받는 많은 량의 통보문들을 배달할수 있다.

제 33 장. 우편봉사기로서의 Linux(Sendmail 능력)

이 장에서는 우편전송대행체 Sendmail 을 리용하여 Linux 를 조직적인 우편봉사기로 전환시키는 방법을 배우게 된다.

우편봉사기에 대한 요구는 망우에서 다중탁상면들을 서로 접속하고 그들에 대한 전자우편봉사를 제공하려고 할 때 제기된다. Sendmail 을 리용하여 Linux 체계를 인터네트로부터 들어 오고 나가는 통보문들은 물론 내부통보문들에 대한 우편봉사기로 동작하도록 구성할수 있다.

이 장에서는 처음에는 우편전송대행체(MTA)의 개념을 보고 그다음 Sendmail, Unix 세계에서 최초의 MTA 와 일부 변화된 MTA 를 훑어 본다. 마지막으로 Sendmail 을 구성하는 업무에 들어 간다.

우편전송대행체의 개념

보통 사용자들이 전자우편체계들에 대하여 생각할 때 그들은 통보문들을 읽고 편집하고 보내는데 리용되는 쏘프트웨어에 대하여 생각한다. 이것은 Netscape 의 우편모듈, Eudora, Pegasus Mail, 혹은 pine 이 될지도 모르지만 모든 경우에 이러한 프로그람들은 단순히 실지 우편물의 경로를 조종하고 배달하는 일을 하는 뒤에 숨어 있는 체계들에 대

한 대면부이다. 이밖의 경우에 사용자의 쏘프트웨어는 실제적으로 나가는 우편물의 배달과 들어 오는 우편물의 접수를 조종한다.

이러한 프로그람들은 우편사용자대행체들(MUAs)이다. 여러가지 MUAs 들이 있는데 그것들의 일감은 사용자에게 전자우편세계에 대한 대면부를 제공하는것이다.

이러한 MUA 들의 뒤면에 우편전송대행체들이 있다. MTAs 들은 우편통보문들을 배달하고 접수하고 경로조종하는 일을 조종한다. 일반적으로 그것들 모두는 인터네트의 전자우편체계(그리고 거의 모든 Unix 에 기초한 망의 전자우편체계)를 구성하는 표준규약(SMTP 로서 알려 진 단일우편전송규약과 같은)을 포함하게 한다.

MUA 와 MTA 사이의 이러한 차이점은 중요한것이다. 조직이나 다른 콤퓨터망을 위한 우편봉사기를 설치하려면 우편전송대행체들을 리용하여 일하는것이 필요하다.

여러가지 MTA 들에 대한 보다 많은 정보는 Linux 전자우편행정관인 HOWTO 에서 리용할수 있으며 http://www.linuxdoc.org/HOWTO/Mail- Administrator -HOWTO.html에서 직결봉사를 리용할수 있다.

일반적인 MTA 로서의 Sendmail

Unix 세계에서 최초의 MTA 가 Sendmail 프로그람이라는것은 거의나 의심할바 없다. 이 프로그람은 누구나 알고 있는 주소화구조(username@some.domain)와 모두가 기대하여 온 특징(자동적인 우편물부치기와 같은)들을 제공하는 인터네트우에서 누구에게나 아주 친숙해 진 우편체계를 정의한다.

그 우월성으로 하여 Sendmail 은 가장 능력 있고 종합적인 Unix MTA 로 되였다. 사실 Sendmail 체계의 구성과 관리는 큰 조직내에서 너무도 복잡해서 Sendmail 체계의 설계, 그것의 구성, 합리적인 운영관리의 주제의 여러책들로 서술되였다.

이장은 Sendmail 의 구성과 사용에 대하여 초점을 둔다. 왜냐하면 이것이 Red Hat Linux 에 포함된 기정 MTA 이고 그것이 Unix 망우에서 아주 공통적이기때문이다.

일러두기 이 장은 Sendmail 에 대한 개괄과 그것을 어떻게 구성하는가를 준다. 화제를 더 깊이 들여 다 보려면 Web 싸이트 **www.sendmail.org** 를 보아야 한다.

그러나 먼저 Linux 에서 리용할수 있는 다른 MTA 들의 일부에 대하여 간단히 개괄하자.

Smail

Sendmail 은 위력한 Unix 우편전송대행체이지만 Linux 세계에서는 smail 이 MTA 로 광범히 리용되고 있다. 그것의 구성은 Sendmail 의 설치보다 더 쉬우며(항상은 아니지만) 직결우편봉사기구성에 아주 적합하게 하는 SMTP 지원과 UUCP 지원을 포함하여 Sendmail 과 류사한 특징들을 가지고 있다.

Red Hat Linux 7.1 은 자기의 기정 MTA 로 Sendmail 을 리용하며 smail 제품을 포함하지 않는다. ftp://ftp.planix.com/pub/Smail/ 으로부터 혹은 http://www.rpmfind.net에서 최신 RPM 을 탐색하여 smail 의 최신판본을 내리적재할수 있다.

Qmail

qmail 은 MTA 경쟁마당에서 자기를 강한 경쟁자로 만드는 여러가지 특징을 가지고 있는 Sendmail 의 변종이다.

qmail 의 작성자는 제품에 대한 우선권을 정하여 안전성을 담보하였다. 오래동안 Sendmail 에서는 비록 그것들이 항상 그후에는 제거되였지만 많은 안전성오유들이 발견되여 왔다. qmail 원리는 우편이 끊임없이 봉사를 실행하고 따라서 해킹시도들을 항시적으로 공개시킴으로써 모든것이 숨은 구멍들을 피할수 있도록 하는것이다.

qmail 은 또한 통보문이 우편통안으로 배달되고 있는 동안에 체계파괴를 일으키는 변질에 대해서 가장 안전한 우편통형식을 리용하여 배달처리를 최대한 믿을수 있게 하려고 하고 있다.

끝으로 qmail 은 전통적인 MTA 들에 속하는 응용프로그람들이 따로따로 가지고 있는 특징들 즉 발송하기,별명붙이기,목록들을 우편으로 보내기,적재관리 등과 같은것을 함께 가지고 있다.

qmail 홈페지는 http://www.qmail.org에 있다. qmail 홈페지에는 사용자가 지원을 받을수 있는 수많은 업무용기업들이 렬거되여 있다.

M4 에 의한 Sendmail 의 구성

이미 언급한바와 같이 Sendmail 의 구성은 지어 숙련된 Unix 체계행정관들에게 있어서도 겁나는 과제일수 있다. 체계의 복잡성을 완전히 정통하는데는 수년이 걸릴수 있다.

Sendmail 의 구성은 /etc/sendmail.cf 에 보관되여 있으며 Red Hat Linux 7.1 과 함께 들어있는 기정구성파일은 구성의 복잡성을 보여 준다.

만일 이 파일들을 들여다 본다면 그것이 Sendmail 의 동태를 정의하는 수많은 극비규칙들을 포함하고 있다는것을 알수 있다. Sendmail 에 대한 전문가가 아닌 그 누구가 기초적인 체계를 구성하려고 하는것은 이런 형태의 구성파일을 수동으로 창조하고 편집하여야 한다면 분별없는것으로 될것이다.

Sendmail 은 구성파일을 만들기 위한 편리를 제공하는데 그것은 M4 라고 하는 프로그람이다. M4 는 훨씬 더 간단히 구성파일들을 만들도록 한다. M4 는 그다음 이러한 파일들을 처리하고 그것을 완전히 쓸수 있는 Sendmail 구성파일들로 변화시킨다. 대체로 지식이 없는 사용자들은 이 방법으로 자기들의 Sendmail 구성을 만들어야 한다.

주의	http://www.sendmail .org/m4/readme.html에서 M4 Sendmail 구성에 대한 론의를 찾을수 있다. M4 를 리용하여 어떻게 Sendmail 을 구성하는가 하는 실례들은 http://www.sendmail .org/m4/intro.html에서 볼수 있다.

이 절은 M4 로써 작업하는데 필요한 Sendmail 구성파일들을 설치하는것으로 시작하고 그다음에 직결우편봉사기의 구성을 본다.

Sendmail 파일의 설치

현재까지 서술한것과 같이 Sendmail 은 세개의 서로 다른 RPM 프로그람묶음으로 나누어져 있다. 그것들은 자기자체의 쏘프트웨어, 구성편의프로그람들 그리고 문서에 대한

것이다. 만일 Sendmail 을 처음으로 사용하거나 구성한다면 세개의 제품들을 모두 설치하려고 할것이다. 이 책에 포함되여 있는 CD-ROM 으로부터 아래의 제품을 설치할수 있다.

```
$ rpm -i /mnt/cdrom/RedHat/RPMS/sendmail-8.11.2-14.i386.rpm
```

Red Hat Linux 7.1 을 설치할 때 Sendmail 구성파일들의 일부만이 기정으로 설치된다. 제품을 설치하는것은 쉽다.이 책에 따라 나온 Red Hat CD-ROM 을 설치하고 두개의 제품들을 설치하도록 rpm 지령을 리용한다.

```
$ rpm -i /mnt/cdrom/RedHat/RPMS/sendmail-8.11.2-14.i386.rpm
$ rpm -i /mnt/cdrom/RedHat/RPMS/sendmail-cf-8.11.2-14.i386.rpm
```

Sendmail 문서파일은 Red Hat Linux 7.1 을 설치할 때 기정으로 설치되지 않는다. 두번째 Red Hat CD-ROM 을 가지고 있지 않는 한(이 책에 포함되여 있지 않다.) ftp.redhat.com 혹은 www.rpmfind.net 와 같은 싸이트로부터 sendmail-8.11.2-14.i386.rpm 제품을 적재하여야 할것이다. /tmp 등록부에 적재하였다고 가정하면 그다음에 아래의 지령으로 그것들을 설치할수 있다.

```
$ rpm -i /tmp/sendmail-doc-8.11.2-14.i386.rpm
```

직결우편봉사기창조

이제는 직결우편봉사기를 위한 구성파일을 볼수 있다. 직결우편봉사기들이 비직결봉사기들보다 구성과 리해가 더 쉽기때문에 이것을 한다.

주의 비록 M4 체계가 Sendmail 구성을 쉽게 해주지만 선택항목들의 수는 여전히 무한히 많다. 구성파일안에서 나타나는 선택항목만을 이 장에서 론의한다. /usr/share/sendmail-cf 등록부의 보조등록부들안에 많은 실례구성파일들이 들어 있다.최종적으로 Sendmail 구성파일을 만들 때 구성파일은 /usr/share/sendmail-cf/cf 등록부안에 속해 있다.

직결우편봉사기는 전용선으로 인터네트에 완전히 련결되여 있는 망용 우편봉사기이다. 망밖에서 누군가가 우편통보문을 국부망우에 있는 사용자에게 직접 보냈을 때 그것은 직접 배달될수 있으며 망우에 있는 사용자가 통보문을 보냈을 때 그것은 직접적으로 송신된다.

이 경우에 들어 오는 통보문들을 사용자의 우편통들에 직접 보내며 사용자들이 보내온 통보문을 직접 배달하도록 Sendmail 을 구성할 필요가 있다. 통보문들은 만일 통보문을 배달하려고 하는 동안에 인터네트접속에서의 고장과 같은 통보문을 배달하는데서 문제가 생기기만 하면 늦어 진 배달은 줄을 서게 된다.

m4 구성파일은 아래와 같이 볼수 있다.

```
include( `../m4/cf.m4')
OSTYPE(`linux')
undefine(` UUCP_RELAY')dnl
undefine(`BITNET_RELAY')dnl
FEATURE(redirect)dnl
FEATURE(a1ways_add_domain)dnl
```

MAILER(1oca1)dn1
MAILER(smtp)

주의 M4 구성파일안에서 괄호에 주의를 돌려야 한다. 첫 괄호는 항상 거꿀외괄호(`)인데 그것은 U.S.건반우에서 Tab 건우에 있다. 다음괄호(')는 보통의 외괄호이다.

행별로 내려 가면서 이것을 따져 보자.

1 행: include(`../m4/cf.m4') 이것들은 Sendmail 구성파일을 만드는데 필요한 일반적인 구성파일들이다.

2 행: OSTYPE(`linux') 적당한 기정값들을 설정하도록 조작체계형을 Linux 로써 정의한다.

3 행: undefine(`UUCP_RELAY') UUCP 중계를 정의하지 않으므로 하여 UUCP 로 관리되는 우편을 접수한 주콤퓨터가 없으며 UUCP 접수자들은 직접 접속되여야 한다는것을 나타내고 있다. UUCP 우편이 대다수의 망들이 직접적으로 접속되여 있지 않는 시기에 설계되였다는것을 고려하여 대다수의 싸이트들은 UUCP_RELAY 를 정의하지 않은대로 남겨 놓을것이다.

4 행: undefine(`BITNET_RELAY') Bitnet 망에 접속되여 있지 않기때문에 이러한 비정의를 남겨 놓았다. 즉 Bitnet 형식을 리용하는 주소들은 동작할수 없을것이다.

5 행: FEATURE(redirect) 설정했다면 address.REDIRECT 주소를 가진 임의의 우편은 거절 당하며 사용자의 새로운 주소를 가리키는 통보문을 내보낸다. 이 방법으로 사용자가 남아 있을 때 자기의 새로운 주소는 .REDIRECT 가 붙은 자기의 낡은 주소로 별명될수 있다.

6 행: FEATURE(always_add_domain) 이 특징은 From 행이 항상 통보문들이 접수자들에 의하여 응답할수 있도록 하기 위하여 국부적인 령역을 포함하는가를 확인하게 한다.

7 행: MAILER(local) 국부적인 우편배달자지원은 Sendmail 이 국부적인 Unix 우편함들에 우편을 배달하도록 한다.

8 행: MAILER(smtp) SMTP 우편배달자지원은 Sendmail 이 통보문들을 직접 접수자우편봉사기에 보내도록 한다.이것은 봉사기가 인터네트에 련결되여서 DNS 봉사들을 할수 있는 체계에서 동작한다.

주의 몇개의 sendmail 행들의 끝에 있는 dnl 은 “delete through newline”의 략칭인데 그것은 보통 아래에서 취급하는 .cf 출구파일안에 있는 빈 행들을 줄인다.

Sendmail 구성파일을 m4 구성파일로부터 만들려면 /usr/lib/sendmail-cf/cf 등록부에 m4 파일을 만들 필요가 있다. 이 경우에 online.mc 파일을 호출하자. 확장자 .mc 은 일반적으로 m4 구성파일들에 대한것과 동일하다.

주의 만일 Sendmail 에 낮이 설다면 **generic-linux.mc** 와 **redhat.mc** 를 비롯하여 이 등록부안에 수많은 실례구성파일들이 들어 있으므로 그것을 **online.mc** 에 복사할수 있다.

다음으로 등록부를 /usr/lib/sendmail-cf/cf 로 변화시키고 아래의 지령을 쓴다.

```
$ m4 online.mc > online.cf
```

이 지령은 m4 를 리용하여 파일을 처리하고 online.cf 라고 하는 Sendmail 구성파일을 발생시킨다.

다음 단계는 존재하는 sendmail.cf 파일의 보관물을 만들고 그것을 이미 창조한것과 함께 놓는것이다. 이것은 root 사용자에 의하여 수행되여야 한다.

```
# cp /etc/sendmail.cf /etc/sendmail.cf.keep
# cp online.cf /etc/sendmail.cf
```

마지막 최종단계는 Sendmail 데몬을 재기동하는것이다.

```
# /etc/rc.d/init.d/sendmail restart
```

데몬조종 편의프로그람들은 만일 우리가 Red Hat Linux 와 다른 배포물을 사용하고 있다면 다른 등록부에 있을수도 있다.

체계시동시에 Sendmail 데몬이 시동되면 그것은 이제는 자기의 새로운 구성파일을 적재할것이다. 만일 Sendmail 을 수동으로 적재하려고 한다면 아래의 지령을 리용한다.

```
# /usr/sbin/sendmail -bd
```

기정으로 Sendmail 데몬은 그것을 무효로 선택하지 않는 한 Linux 의 거의 모든 배포물들에서 체계시동시에 적재한다. 만일 Sendmail 을 자기의 boot 덩어리에 추가하려고 한다면 rc.local 파일에 /usr/sbin/sendmail -bd 지령을 리용할수 있다.

일러두기 만일 큰 Sendmail 싸이트를 실행하려면 도서《*Sendmail* by Costales and Allman from O'Reilly & Associates (1997. 2판)》을 보면 잘 알수 있다.

▶미리 알아두기

이 장은 Linux 를 리용하여 자기자체로 우편봉사기를 설치하는 기초적인것들을 파악하는것으로써 이 책의 기본부분을 이루고 있다. 수중에 있는 이러한 지식과 함께 현재까지 Linux 체계들을 리용하고 구성하고 관리하여온 경험에 의하여 Linux 를 리용해서 해야 할 모든 콤퓨터과제를 수행할수 있어야 한다.

이 책의 나머지부분은 여러가지 부록들로 이루어져있는데 대부분이 Linux 를 매일 리용하는데 도움이 될것이다. 개별적으로《Linux 지령참고서》(부록 3)는 그 대부분이 정연한 기초로써 리용될 Linux 의 주요 지령들의 문법에 대한 고속안내를 제공한다.

더우기 부록 2《Linux 정보원천》은 Linux 지식과 수완을 넓히는데 쓸모 있는 자원들을 제공한다. Linux 는 빨리 정통할수 없는 거대한 대지이다.지어 가장 능숙한 Linux 전문가들도 거의 매일이다싶이 새로운 요령들과 비밀들,수완들을 배우고 있다. 그렇다고 해서 두려워해서는 안된다. 더 중요한것은 Linux 의 새로운 그룹들에 혹은 대중적인 Linux 응용프로그람작성자들에게 물어 보는것을 두려워 하지 말아야 한다. http://www.linuxdoc.org에서 HOWTO 들과 같은 기초적인 문서들을 보다 오래동안 읽은 저자들과 다른 전문사용자들은 보통 풋내기 Linux 사용자들에게 안내해 주고 통찰력을 키워 주며 충고를 주는것을 더 행복하게 여긴다. 결국 자라나고 있는 Linux 사용자들의 집단에 대하여 지식을 주고 인식을 바로 하게 하며 수완을 키워 주는것은 Linux 개발자들과 사용자들의 관심속에 있다.

부 록

부록 1. 세계적인 Linux(비영어 Linux 배포물)

Linux는 항상 국제적인 관심사로 되여 왔다. 핀란드로부터의 핵심부, 도이췰란드로부터의 GUI, 무료배포를 조장시키는 미국으로부터의 관례를 벗어 난 사용허가에 의하여 Linux는 세계적범위로 퍼져 나갔다. 일부 나라들에서는 개발자들이 Linux를 자체의 민족어로 번역하였다. 대다수의 집단들이 기본배포물에 기초하여 자체번역하였다.

이 책은 영어저자들이 쓴것이므로 아래에서 서술한 비영어 배포물의 질에 대해서는 주장하지 않는다.

흐르바쯔까어 Linux

MicroLinux

MicroLinux는 DOS구획에 앉힐수 있게 설계된 작은 흐르바쯔까어배포물이다. 이것은 Windows 95/98/Me체계들을 재구획화할수 없는 사용자들에게 편리하게 만들어 졌으며 Windows(혹은 DOS)와 Linux를 둘다 실행할것을 요구한다.

HULK-Hrvatska udruga Linux korisnika

WWW:http://linux.hr/

프랑스어 Linux

MNIS Linux

MNIS는 MNIS Linux로 알려진 Linux의 프랑스어배포물을 만든다. MNIS Linux는 PC에 기초한 X워크스테이션으로 만들었는데 K탁상환경과 같은 특수한 특성을 가지고 있다. 또한 자기의 배포물에서 직접 Window 95/NT/2000 응용프로그람을 실행할수 있는 제품화된 X-Win32판매품을 가지고 있다.

MNIS

57, rue d'Amsterdam

75008 Paris

France

E-mail: mnis@mnis.fr

WWW: http: //www.mnis.fr/

도이취어 Linux

EasyLinux

EasyLinux 는 Linux 를 사용하지 않는 사람들을 위하여 설계된 도이취어에 기초한 Linux 배포물이다. 여기에는 필수적인 대다수 Linux 구성파일들을 위한 많은 도형방식구성편의프로그람들이 있다.

easy Information Technology GmbH

Am Seerhein 8

78467 Konstanz

Germany

WWW: http://www.easyLinux.com/

Icepack Linux

Icepack Linux 는 Linux 를 사용하지 않는 사람들을 위하여 설계된 도이취어에 기초한 Linux 배포물이다. 비록 대부분이 GNOME 와 KDE 와 같은 제품들을 리용하지만 그것은 다른 Linux 배포물들과는 독자적으로 개발된것이다.

Icepack Linux

E-mail: support@icepack-linux.com

WWW: http://www.icepack-linux.com/

일본어 Linux

Laser5 Linux

Laser5 Linux 는 Red Hat Linux 에 기초한 일본어배포물이다. 이것도 역시 Linux 워크스테이션, 봉사기, 클라스터화된 체계들을 가지고 있다.

LASER5 Co.,Ltd.

NOV Building,3F,3-10-7 Yushima

Bunkyo-ku

Tokyo

Japan

113-0034

E-mail: info@laser5.co.jp

WWW: http://www.laser5.co.jp/

Vine Linux

Vine Linux는 또 다른 일본어배포물이다. 이것의 최신판인 Vine Linux 2.1은 Red Hat Linux 6.2에 기초한것이다. Vine Linux는 많은 일본어서체들과 응용프로그람들을 포함한다. 그것은 자기의 일본어 Web싸이트우에서 Red Hat Linux에 의하여 팔리고 있다.

Vine Linux

WWW:http://www.vinelinux.org/

뽀르뚜갈어 Linux

Conectiva Linux

Conectiva Linux는 브라질에 있는 Conectiva Informatica에 의하여 만들어 진 Red Hat Linux의 뽀르뚜갈어판본이다. 현 시점에서 Conectiva Linux의 현행배포물은 6판인데 여기에는 KDE 2판과 XFree4.01판이 있다.

Conectiva Informatica

Rua Tocantins,89 - Cristo Rei

Zip: 80050-430

Curitiba(PR)

Brazil

E-mail: info@conectiva.com.br

WWW: http://www.conectiva.com.br/

로씨야어 Linux

Open Kernel:Linux

UrbanSoft의 Kernel:Linux은 Red Hat Linux에 기초한 로씨야어배포물인데 그것은 로씨야어서체, 로씨야어 X Windows환경, 표준Red Hat환경에는 없는 추가적인 로씨야어제품을 제공한다. 현행판본은 Red Hat 6.2로부터 파생된것이다.

UrbanSoft

E-mail:info@usoft.spb.ru

WWW:http://www.usoft.spb.ru/

에스빠냐어 Linux

Eurielec Linux

Eurielec Linux 는 현재 에스빠냐에서 Red Hat Linux 에 기초한 또 다른 비영어 Linux 배포물이다. Eurielec Linux 의 현행판본은 Red Hat Linux 6.2 에 기초하고 있다.

Escuela Tecnica Superior Ingenieros de Telecomunicacion

Avda. Complutense s/n - 28040

Madrid

Spain

E-mail: linux@eurielec.etsit.upm.es

WWW: http://www.eurielec.etsit.upm.es/linux/

ESware Linux

ESware Linux 는 설치, 지령행, 문서, 사용자설명서를 비롯하여 모든것이 에스빠냐어로 된 비영어 Linux 배포물이다.

ESWARE LINUX

Procion 1-3

Edif.Oficor

28023 La Florida

Madrid

Spain

WWW:http://www.esware.com/

스웨리예어 Linux

Blue Linux

Blue Linux 는 Debian Linux 2.2 에 기초한 스웨리예어배포물이다. Linux 직결봉사에 의하면 그것은 하드디스크에 설치되지 못한다.

Blue Linux

WWW:http://blue.swt.nu/

뛰르끼예어 Linux

Gelecek Linux

Gelecek Linux는 현재 뛰르끼예에서 Red Hat Linux에 기초하고 있는 또 다른 비영어 Linux배포물이다.

Gelecek Bilisim ve Iletisim A.S

Perpa Ticaret Merkezi

B Blok Kat: 11 No: 1932

Sisli,Istanbul

Turkey

E-mail: info@gelecek.com.tr

WWW: http://www.gelecek.com.tr/

다언어 Linux

대다수의 주요배포물들은 적어도 여러가지 서로 다른 언어로 된 설치프로그람판본들을 가지고 있다. 여기에는 한가지이상의 언어(비영어)를 쓸수 있는 여러가지 Linux배포물들이 포함되여 있다.

Best Linux

Best Linux는 핀란드어에 기초한 독자적인 배포물이다. 그의 현행제품인 Best Linux 2000은 《우리가 매일 리용하는 대부분의 쏘프트웨어들을 포함하고 있는 KDE의 탁상과 비슷한 도형방식의 창문》에 직접 설치하도록 설계된것이다. Best Linux는 핀란드어, 스웨리예어, 영어, 로씨야어, 에스또니아어, 프랑스어, 도이취어, 에스빠냐어에 대한 지원기능을 가진다. 또한 부분적으로 다른 14개 유럽언어들에 대한 지원기능을 가진다.

SOT Finnish Software Engineering Ltd.

Hermiankatu 8 E

Fin-33720 Tampere

Finland

E-mail: finland@bestlinux.net

WWW: http://www.bestlinux.net/

Caldera Linux

Caldera는 자기 배포물의 판본을 많은 주요언어로 개발하여 왔다. 자기의 Web싸이트에 의하면 Caldera는 도이취어, 프랑스어, 이딸리아어, 일본어, 조선어 등을 포함하여 여러가지 서로 다른 언어로 OpenLinux의 기술적인 지원을 보장한다.

Caldera Systems, Inc.

240 West Center Street

Orem,Utah 84057

U.S.A.

E-mail: info@calderasystems.com

WWW: http://www.caldera.com/

Debian Linux

Debian Linux 는 지원자들에 의하여 개발되였다. 대다수의 지원자들은 Debian Linux를 다른 언어로 번역하려고 노력하고 있다. 현 시점에서 문서와 지원을 어느 정도 부분적으로 수정한 17개의 서로 다른 언어로 쓸수 있다. 그것이 지원자들에 의하여 개발되기때문에 이 배포물에 대한 하나의 지원쎈터가 없다.

E-mail: debian-project@lists.debian.org

WWW: http://www.Debian.org/

Mandrake Linux

Mandrake Linux(Linux-Mandrake 로도 알려 졌다.)은 보충적인 언어를 비롯하여 추가적인 특성들을 지원하도록 확장된 Red Hat Linux로부터 개발되였다.

MandrakeSoft Inc.

2400 N.Lincoln Ave.

Altadena,CA 91001

U.S.A.

E-mail: sales@mandrakesoft.com

WWW:http://www.linux-mandrake.com/

Red Hat Linux

Red Hat 는 많은 주요언어들로 자기 배포물의 판본들을 개발하여 왔다. Red Hat Linux 7.1의 완전한 판본과 함께 제공되는 Linux문서 CD-ROM은 프랑스어, 영어, 도이취어, 이딸리아어와 에스빠냐어로 된 문서를 포함한다. Red Hat Linux 7.1 설치프로

그람은 19개의 서로 다른 유럽언어들과 일본어로 Linux를 설치하도록 한다.

Red Hat, Inc.

2600 Meridian Parkway

Durham, NC 27713

U.S.A.

E-mail:See http://www.redhat.com/about/contact/directory.html

for a list of e-mail contacts.

WWW: http://www.redhat.com/

S.u.S.E Linux

도이취어에 기초한것으로서 S.u.S.E는 국제적인 경쟁마당에서 Linux 지휘자로 될지도 모른다. 그것의 지원자료기지는 프랑스어, 영어, 도이취어, 이딸리아어와 에스빠냐어로 쓸수 있다. 그것의 설치프로그람은 많은 유럽언어들에 대한 지원기능을 가지고 있다.

S.u.S.E.

Schanzackerstrafse 10

90443 Nurmberg

Germany

E-mail: suse@suse.de

WWW: http://www.suse.de/

혹은

S.u.S.E.

580 Second Street Suite 210

Oakland, CA 94607

U.S.A.

E-mail: info@suse.com

WWW: http://www.suse.com/

TurboLinux

TurboLinux는 일본어와 중어로 쓸수 있다. TurboLinux의 매 언어에 따르는 판본에는 열람기와 탁상과 같은 언어전용도구들이 있다.

TurboLinux
8000 Marina Boulevard
Suite 300
Brisbane, CA 94005
E-mail: support@turbolinux.com
WWW: http://www.turbolinux.com/

부록 2. Linux 정보원천

이 부록은 Linux 와 관련된 추가적인 정보를 찾을수 있고 Linux 배포물들을 내리적재하거나 구입할수 있으며 도움을 청할수 있는 인터네트자원들을 서술한다.

일반정보

JustLinux	http://www.justlinux.com/
LinuxAdministrator'sSecurityGuide	http://www.linuxdoc.org/LDP/lasg/
Linux Documentation Project	http: //www. linuxdoc.org/
Linux Frequently Asked Questions with Answers	http: //www. mainmatter.corn/linux-faq/toc.html
Linux Hardware	http://www.linuxhardware.net/
LinuxInstallationandGettingStarted	http://www.linuxdoc.org/LDP/gs/gs.html
Linux International	http://www.li.org/
LinuxJournal	http://www.linuxjournal.com/
LinuxNOW!	http://www.linuxnow.com/
Linux Online	http://www.linux.org/
LinuxSystemAdministrators'Guide	http://www.linuxdoc.org/LDP/sag/index.html
LinuxToday	http://www.linuxtoday.com/
LinuxUserGroupRegistry	http://www.linux.org/users/index.html

The Linux Web Ring	http://1inuxwebring .org/
Slashdot	http://www.s1ashdot.org/
TheNetworkAdministrators'Guide	http://www.1inuxdoc.org/LDP/ nag2/index.htm1
WovenGoodsforLinux	http://www.fokus.gmd.de/1inux/

Linux 전문자격증수여기관

Digital Metrics	http://www.digitalmetrics.corn/
Linux Professional Institute	http: //www. 1pi. org/
Red Hat Certified Engineer	http://www. redhat. corn/ training/rhce/courses/
Sair GNU/Linux	http://www.linuxcertification org/

Linux 배포물

BestLinux	http://www.bestlinux.net/
BlackCatLinux	http://www.b1ackcat1inux.com/
Blue Linux	http://b1ue.swt.nu/
CalderaOpenLinux	http://www.ca1derasystems.com/
ConectivaLinux(BrazilianPortuguese)	http://www.conectiva.com.br/
Corel Linux	http://1inux.core1.com/
Debian/GNULinux	http://www.debian.org/
DLXLinux	http://www.wu-wien.ac.at/usr/
h93/h9301726/d1x.htm1	
EasyLinux	http://www.easylinux.corn/
ESware Linux	http://www.esware.com/
EurielecLinux(Spanish)	http://www.eurielec.etsit .upm.es/linux/
Gelecek Linux	http://www. gelecek.corn.tr/

Icepack Linux	http://www.icepack-1inux.corn/
Laser5 Linux	http://www.1aser5.co.jp/
Linux Mandrake	http://www.1inux-mandrake.com/
LinuxPPC (for PowerPC chips)	http://www.1inuxppc.org/
LinuxWare	http://www.trans-am.corn/1inux.htm
MkLinux (for the Power Macintosh)	http://www.mklinux.org/
MNIS Linux (French)	http://www.mnis.fr/
Red Hat Linux	http://www.redhat.com/
Slackware Linux	http://www.s1ackware.com/
Stampede Linux	http://www.stampede.org/
S.u.S.E.Linux(German and English)	http://www.suse.com/
TurboLinux	http://www.turbo1inux.com/
Vine Linux	http://www.vine1inux.org/

우편목록과 소식그룹

alt.os.linux.best	news:a1t.os.1inux.best
alt.os.linux.caldera	news:att.os.linux.caldera
alt.os.linux.corel	news:a1t.os.1inux.corel
alt.os.linux.dial-up	news:alt. os.1inux.dial-up
alt.os.linux.mandrake	news:alt. os.1inux.mandrake
alt.os.linux.redhat	news:a1t.os.1inux.redhat
alt.os.linux.slackware	news:a1t.os.1i nux.slackware
alt.os.linux.storm	news:a1t.os.1inux.storm
alt.os.linux.suse	news:a1t.os.1inux.suse

alt.os.liux.turbolinux	news:alt.os.linux.turbolinux
comp.os.linux.admin	news:comp.os.linux.admin
comp.os.linux.advocacy	news comp.os.linux.advocacy
comp.os.linux.alpha	news:comp.os.1inux.alpha
comp.os.linux.announce	news:comp.os.linux.announce
comp.os.linux.development.apps	news:comp.os.linux .development.apps
comp.os.linux.development.system	news:comp.os.1inux .development.system
cornp.os.linux.embedded	news:comp.os.1inux.embedded
comp.os.linux.hardware	news:comp.os.1inux.hardware
comp.os.linux.help	news:comp.os.1inux.he1p
comp.os.linux.m68k	news:comp.os.1inux.m68k
comp.os.linux.misc	news:comp.os.1inux.misc
comp.os.linux.networking	news:comp.os.1inux.networking
comp.os.linux.portable	news:comp.os.1inux.portable
comp.os.linux.powerpc	news:comp.os.1inux.powerpc
comp.os.linux.questions	news:comp.os.1inux.questions
comp.os.linux.redhat	news:comp.os.1inux.redhat
comp.os.linux.security	news:comp.os.1inux.security
comp.os.linux.setup	news:comp.os.1inux.setup
comp.os.linux.x	**news:comp.os.1inux.x**
Linux Mailing Lists	http://www.1inuxlookup.com/ html/main/mai1inglists.html

비인텔가동환경

LinuxAlphaFTPArchive	http://gatekeeper.dec.com/pub/ DEC/Linux-Alpha/
LinuxforBeBox	http://www.sowerbutts.com/ belinux/introduction.html
Linuxfor Sparc Processors	http://www.ultralinux.org/
Linux/m68kforMacintosh	http://www.mac.linux-m68k.org/
TheLinux/m68kHome Pages	http://www.linux-m68k.org/
MKLinuxArchive	http://ftp.sunet.se/pub/os/ Linux/mklinux/mklinux/
SGI-Linux	http://linus.linux.sgi.com/

비영어 싸이트

EPCHomePage(Spanish)	http://www.arrakis.es/ ~epujol/linux/
Linux (Czech)	http://www.linux.cz/
Linux-VielUnix fur wenig Geld (German)	http://www.uni-tuebingen.de/ zdv/projekte/linux/
Linux Indonesia (Indonesian)	http://www.linux.or.id/

Linux CD-ROM 의 구입장소

CD-ROMShop	http://www.cdromshop.com/
CheapBytes	http://www.cheapbytes.com/
InfoMagic	http://www.infomagic.com/
Linux Central	http://linuxcentral.com/
LinuxMall	http://www.linuxmall.com/

Linux 와 Linux 파일의 내리적재장소

BLINUX's FTP Site	ftp://leb.net/pub/blinux/
Caldera'sFTPSite	ftp://ftp.caldera.com/pub/
Debian's FTP Site	ftp://ftp.debian.org/
Freshmeat.net	http://www.freshmeat.net/
GeorgiaTech'sLinuxArchive	ftp://ftp.cc.gatech.edu/ pub/linux/
LinuxApplications	**http://www.linuxapps.com/**
Linux Kernel Archives	ftp://ftp.kernel.org/
LinuxSoftwareMap	http://www.execpc.com/lsm/
RedHat'sFTPSite	ftp://ftp.redhat.com/pub/
Tucows Linux	http://linux.tucows.com/
UNC'sLinuxArchive	http://www.ibiblio.org/pub/ Linux/
WovenGoodsforLinux'sSoftwarePage	http://www.fokus.gmd.de/linux/ linux-softw.html

X Windows

MetroLink	http://www.metrolink.com/
XFree86 Project	http://www.xfree86.org/
Xi Graphics	http://www.xinside.com/

부록 3. Linux 지령참고서

이 부록은 대부분의 Red Hat Linux 가 설치된 다음의 등록부에서 보게 되는 일반 Linux 지령들과 봉사프로그람들의 사용법에 대하여 소개한다.

- /bin
- /sbin
- /usr/bin
- /usr/sbin

여기서 기본은 많은 주요 Linux 지령들의 문법과 일반선택항목에 대하여 빨리 참고할수 있게 하는것이다. 이 지령들에 대한 보다 구체적인 설명과 여기서 서술하지 않은 지령들은 일반적으로 지령에 대한 사용설명서페지(man 지령에 의하여 호출되는)에서 찾아 볼수 있다.

이 부록은 결코 매 Linux 지령들의 모든 선택항목, 기발과 인수들에 대한 완전한 안내서로는 되지 못한다. 대신 일반적이고도 중요한 지령들의 사용법을 아주 일반적인 형태로 줄수 있게 설계되였다. 지령들에 대한 보다 더 구체적인것은 지령과 관련된 사용설명서페지에서 방조를 받을수 있다.

이 부록은 표준 Linux man 페지와 같은 형식으로 지령들의 종합적인 문법을 소개하였으며 지령에 대한 설명을 주고 뒤따라 주요기발과 인수들에 대한 설명을 주었다.

이 부록을 알맞춤하게 만들기 위하여 대부분의 호상작용하는 프로그람, 데몬, Red Hat 특수지령들은 넣지 않았다. 그래도 이 부록에는 근 200여개의 지령들이 서술되여 있다.

지령들은 영어자모순으로 배렬되였다.

이 부록에 서술된 지령

이 참고서에는 다음의 지령들이 서술되였다.

arch	atrm	biff	chgrp
at	badblocks	cal	chkconfig
atd	batch	cat	chmod
atq	bc	checkalias	chown
clear	grep	listalias	mkdir
compress	roupadd	ln	mkdosfs
cp	groupdel	loadkeys	mke2fs
crontab	roupmod	logger	mkfs
cryptdir	roups	login	mkpasswd
date	grpck	logname	mkswap
dc	gunzip	logrotate	mlabel
decryptdir	gzexe	lpd	mmd
depmod	zip	lpq	mmove
df	alt	lpr	modprobe
dir	head	lprm	more
dmesg	hostname	ls	mount
dnsdomainname	id	lsdev	mrd

dnsquery	ifconfig	lsmod	mren
domainname	ifdown	mail	mtype
du	ifport	mailq	mv
e2fsck	ifup	mailto	netstat
echo	insmod	man	newgrp
egrep	kbd_mode	mattrib	newusers
ealse	kbdrate	mbadblocks	nisdomainname
fdisk	kill	mcd	nslookup
fgrep	killall	mcopy	passwd
file	ksyms	mdel	paste
find	last	mdeltree	pathchk
finger	ldd	mdir	pidof
free	less	messages	ping
gpasswd	lilo	mformat	pppstats
ps	route	tar	uudecode
pwck	rsh	timeconfig	uuencode
pwconv	runlevel	timed	vipw
pwd	rup	timedc	vmstat
pwunconv	rusers	top	w
quota	rwho	touch	wc
quotacheck	rwhod	traceroute	whereis
quotaoff	setclock	true	which
quotaon	setkeycodes	umount	whoami
rcp	showkey	uname	ypdomainname
rdate	showmount	uncompress	zcat
rdist	shutdown	unzip	zgrep
repquota	sort	uptime	zip
rlogin	statserial	useradd	zipgrep
rm	su	userdel	zipinfo
rmdir	sync	usermod	zmore
rmmod	tail	users	znew

arch

설명 — Linux가 실행되는 콤퓨터의 방식을 현시한다. 실례로 펜티움에 기초한 체계는 **i586** 으로, 80486 에 기초한 콤퓨터는 **i486** 으로, Alpha 에 기초한 콤퓨터는 **axp** 로 현시한다.

문법 — arch

주요기발과 선택항목 — 없음

at

설명 — 특정한 시간에 실행해야 할 지령의 일정을 작성한다. 사용자별 지령을 입력하거나 파일로부터 지령을 제공 받는다. 매개 일감은 일정작성대기렬에 추가되며 일감번호가 주어 진다.

문법 — at[-q **렬**][-f **파일이름**][-l][-m][-d **일감**[**일감** ...]] TIME

주요기발과 선택항목

- -d **일감**: 일감번호에 의하여 지정된 일감을 삭제한다.
- -f **파일이름**: 지령을 입력하는것이 아니라 지정된 파일로부터 일정작성된 지령들을 읽는다.
- -l: 일정작성대기렬의 지령들을 현시한다(TIME 인수는 무시된다.).
- -m: 일감이 끝나자마자 일감일정작성자에게 전자우편을 보내며 통보문을 출구한다.
- -q **렬**: 처리하려는 대기렬을 지정한다. 대기렬은 단일문자(a-z 혹은 A-Z)로 지정된다. 기정으로 대기렬은 at 지령과 그와 관련되는 지령에 쓰인다.

주의 — 시간지정: 시간지정을 위한 여러가지 선택항목들이 있는데 그것들은 다음과 같다.

- *HH:MM* 은 11:15 혹은 22:30 과 같이 시간과 분을 지정한다. 11:15AM 혹은 10:30PM 과 같이 뒤에 AM 과 PM 을 붙이기도 한다.
- midnight(24:00 혹은 12:00PM), noon(12:00) 과 teatime1(16:00)은 규정된 시간을 지정하는 예정된 단어이다.
- *MMDDYY*, *MM/DD/YY*, *DD.MM.YY* 는 022598 혹은 25.02.98 과 같이 지정한 날자를 가리킬 때 리용할 수 있다.
- now 는 현재시간을 지정한다.
- tomorrow 는 다음날을 지정한다.

- +를 리용하여 지정된 시간으로부터 minutes, hours, days, weeks 편차를 지정한다. 실례로 현재날로부터 2일후 정오에 실행할 지령일정을 작성하기 위해서는 noon + 2 days를 리용할수 있다.

atd

설명

at와 batch지령과 같이 프로그람에 의하여 제일 마지막에 실행되도록 일정이 작성된 일감을 실행하는 데몬이다.

문법

atd [-l **적재**] [-b **간격**]

주요기발과 선택항목

- -b **간격**: 두 묶음일감들의 시작사이 최소간격을 초단위로 지정한다. 기정값은 60초이다.
- -l **적재**: 일정이 작성된 묶음일감들이 실행되지 않는 적재한계를 지정한다. 기정값은 0.8이다.

atq

설명

일정작성대기렬에 있는 at지령에 의하여 일정이 작성된 일감들을 현시한다. 이것은 at -l과 같다.

문법

atq [-q **렬**]

주요기발과 선택항목

- -q **렬**: 처리하려는 대기렬을 지정한다. 대기렬은 단일문자(a-z 혹은 A-Z)로 지정된다. 기정으로 대기렬은 at나 그와 관련된 지령들에 쓰인다.

atrm

설명

at 일정작성대기렬로부터 지정된 일감을 삭제한다. 이것은 at -d와 같다.

문법

atrm **일감** [**일감** ...]

주요기발과 선택항목

없음

badblocks

설명

장치(일반적으로 하드디스크)안의 나쁜 블로크를 검사한다.

문법

badblocks [-o **파일이름**] [-w] device **블로크수**

주요기발과 선택항목

- -o **파일이름**: 표준출구대신에 결과를 써넣을 파일을 지정한다.

- -w: 읽기전용검사대신에 쓰기검사를 사용한다. 즉 장치의 매개 블로크에 자료를 써넣은 다음 블로크로부터 다시 읽는다.

주의

/dev/hda2 나 /dev/sdb3 과 같이 완전한 Linux 장치경로를 써서 장치를 지정해야 한다. 장치에 대한 블로크번호는 중요하다(이 정보는 fdisk 로 결정한다).

경고

-w 기발은 중요한 자료가 있는 장치에 대해서는 쓰지 말아야 한다. -w기발로 검사되는 장치에 보관된 자료는 검사과정에 지워 진다.

batch

설명

체계적재준위가 허용되는동안 지정된 시간에 실행되여야할 지령의 일정을 작성한다. 사용자는 지령을 넣어 주거나 파일로부터 지령을 제공 받을수 있다. 매 일감은 일정작성대기렬에 추가되며 여기에는 일감번호가 붙는다.

문법

batch [-q **렬**] [-f **파일이름**] [-m] TIME

주요기발과 선택항목

- -f **파일이름**: 지령을 재촉문에 입력하는것이 아니라 지정된 파일로부터 일정작성된 지령을 읽는다.
- -m: 일감이 끝나자마자 일감작성자에게 전자우편을 보내며 통보문을 출구한다.
- -q **렬**: 처리하려는 대기렬을 지정한다. 대기렬은 단일문자(a-z 혹은 A-Z)로 지정된다. 기정적으로 b 렬은 batch 에 리용된다.

주의

시간지정하기: 시간을 설정하는 방식에는 여러가지가 있는데 이것들은 다음과 같다.

- *HH:MM* 은 11:15 혹은 22:30 과 같이 시간과 분을 지정한다. 11;15AM 혹은 22:30PM 과 같이 AM 과 PM 을 뒤에 붙인다.
- midnight(24:00 혹은 12:00PM), noon(12:00) 과 teatime(16:00)은 지정한 시간을 가리키는 단어이다.
- *MMDDYY*, *MM/DD/YY*, *DD.MM.YY* 는 022598 혹은 25.02.98 과 같이 지정한 날자를 가리킨다.
- now 는 현재시간을 가리킨다.
- tomorrow 는 다음날을 지정한다.
- +는 지정한 시간으로부터 minutes, hours, days, weeks 의 편차를 지정한다. 실례로 현재날로부터 2일

후 정오에 실행할 지령은 noon + 2 days 로 작성할수 있다.

bc

설명 | 사용자가 재촉문에 평가식을 줄수도 있고 지정된 파일의 모든 식을 처리할수도 있다.

문법 | bc [**파일이름 ...**]

주요기발과 선택항목 | bc 에서 리용하는 식들의 문법은 대체로 c 프로그람작성언어에 기초하고 있다. 구체적인것은 bc 사용설명서페지를 참고하여야 한다.
인수로 제공된 파일의 식은 사용자가 처리하려는 추가적인 식을 넣기 위한 재촉문으로 돌아 가기전에 처리한다.

biff

설명 | 새로운 우편이 도착하면 사용자에게 통보하고 통보문을 누가 보내주었는가를 알려 준다.

문법 | biff [ny]

주요기발과 선택항목

- n: 설정되면 우편도착통보를 하지 않는다.
- y: 설정되면 우편도착통보를 한다.

cal

설명 | 월이나 년으로 력서를 현시한다. 월이나 년을 지정하지 않으면 현재 월을 현시한다.

문법 | cal [-j] [-y] [**월** [**년**]]

주요기발과 선택항목

- [-j]: 그리고리언날자가 아니라 율리언날자로 현시한다.
- [-y]: 월력이 아니라 년력을 현시한다.

주의 | 인수가 하나인 경우 그 수자는 년을 현시하는데 1 부터 9999 사이 값으로 설정한다(년은 완전한 형식으로 써야 한다. 즉 98 이 아니라 1998 로 주어야 한다). 인수가 두개인 경우 첫번째 인수는 월을 지정하는데 그 값은 1 부터 12 까지이며 두번째 인수는 1 부터 9999 까지 값으로 년을 지정한다.

cat

설명 | 하나이상의 파일들을 련결해서 표준출구로 현시한다. 파일이 주어 지지 않으면 표준입구가 직접 표준출구에로 출구된다.

문법 cat [--benstvAET] [--number] [--number-nonblank] [--squeeze-blank] [--show-nonprinting] [--show-ends] [--show-tabs] [--show-all] [**파일 ...**]

주요기발과 선택항목

- -A/--show-all: 매 행의 끝에 $를 현시하고 tab문자는 ^l로 현시하며 조종문자는 앞에 ^을 붙인다. 이것은 -v, -T와 -E의 결합형식과 같다.
- -b/--number-nonblank: 공백이 아닌 모든 행들에 번호를 붙인다. 번호는 1부터 시작한다.
- -e: 매 행의 끝에 $를 현시하며 조종문자의 앞에 ^을 붙인다. 이것은 -V와 -E의 결합형식과 같다.
- -E/--show-ends: 매 행의 끝에 $를 현시한다.
- -n/--number: 모든 행에 번호를 붙인다. 번호는 1부터 시작된다.
- -s/--squeeze-blank: 화면에 공백행이 련속으로 나타날 때 한개의 공백으로 바꾼다.
- -t: tab문자를 ^l로 현시하며 조종문자앞에 ^을 붙인다. 이것은 -v와 -T의 결합형식과 같다.
- -T/--show-tabs: tab문자를 ^l로 현시한다.
- -v/--show-nonprinting: a앞에 ^이 붙은 조종문자를 현시한다.

checkalias

설명 사용자의 파일을 검사한 다음 지정된 별명이 정의되였는가를 보기 위하여 체계별명파일을 검사한다.

문법 checkalias **별명** [**별명 ...**]

주요기발과 선택항목 없음

chgrp

설명 하나이상의 파일이나 등록부의 그룹소유권을 변경시킨다.

문법 chgrp [-Rcfv] [--recursive] [--changes] [--silent] [--quiet] [--verbose] **그룹 파일이름 ...**

주요기발과 선택항목

- -c/--changes: 소유권이 변경되는 파일이름만을 현시한다.
- -f/--silent/--quiet: 파일의 소유권을 변화시킬수 없으

면 오유통보문의 현시를 금지시킨다.

- -R/--recursive: 변경된 소유권을 가진 임의의 등록부의 모든 보조등록부안에 있는 모든 파일들의 소유권을 차례로 변경시킨다.
- -v/--verbose: 모든 소유권변화결과를 현시한다.

주의　　그룹은 이름이나 그룹 ID 로 구별할수 있다.

chkconfig

설명　　체계실행준위에 대한 설정을 조종하거나 현시한다.

문법

chkconfig -list[**이름**]

chkconfig -add **이름**

chkconfig -del **이름**

chkconfig <on l off l reset>[--level **준위들**]

chkconfig [--level **준위들**] name

주요기발과 선택항목

- --add **이름**: chkconfig 로 관리기를 위한 새로운 봉사기를 추가하고 필요한 시동과 닫기항목들이 존재하는가를 검사한다. 항목이 없으면 그것을 작성해야 한다.
- -del **이름**: 관리기로부터 이름이 붙은 봉사기를 삭제하며 그와 련관된것들도 역시 삭제한다.
- --level[**준위들**]: 이름이 붙은 봉사기가 속하는 실행준위를 수값으로 지정한다.
- -list **이름**: chkconfig 가 알려 준 모든 봉사기들과 그것들과 관련된 임의의 정보들을 기입한다. 이름이 붙은 봉사기가 지정되면 현시된 봉사기에 대해서 정보를 제시한다.
- off: 후에 봉사기이름이 지정되면 지정된 실행준위를 위한 봉사기상태를 정지상태로 변화시킨다.실행준위가 지정되지 않으면 이 선택항목은 실행준위 3, 4 와 5 에 영향을 준다.
- on: 후에 봉사기이름이 지정되면 지정된 실행준위를 위한 봉사기의 상태를 시동상태로 변화시킨다. 실행준위가 지정되지 않으면 이 선택항목은 실행준위 3, 4 와 5 에 영향을 준다.
- reset: 후에 봉사기이름이 지정되면 지정된 실행준위

를 위한 봉사기의 상태를 init 스크립트로 지정하는 표준상태로 설정한다. 그러면 이 선택항목은 실행준위를 설정한다.

chmod

설명
하나이상의 파일이나 등록부의 호출권을 변화시킨다.

문법
chmod [-Rcfv] [--recursive] [--changes] [--silent] [--quiet] [--verbose] **방식 파일 ...**

주요기발과 선택항목

- -c/--changes: 허가가 변화되는 파일이름만을 현시한다.
- -f/--silent/--quiet: 파일의 허가를 변경시킬수 없으면 오유통보문의 현시를 금지시킨다.
- -R/--recursive: 변화된 허가를 가지는 등록부의 모든 보조등록부안에 있는 모든 파일들의 허가를 반복하여 변화시킨다.
- -v/--verbose: 모든 허가변화결과를 현시한다.

주의
방식들을 두가지 방법 즉 기호와 수로 지정할수 있다. 기호로 지정될 때 방식은 [ugoa][[+-=] [rwxXstugo...] 형태를 취한다. 첫번째 요소([ugoa])는 허가를 받는 사용자를 현시한다(u는 파일이나 등록부를 소유하는 사용자, g는 파일이나 등록부를 소유한 그룹의 모든 성원, o 는 소유자나 소유자그룹이 아닌 임의의 사용자, a 는 모든 사용자). +기호는 특수한 허가를 첨가하는 방식을 의미하고 -기호는 이미 존재하는 허가를 제거하는 방식을 의미한다. =기호는 현재허가를 바꾼다는것을 의미한다. 방식의 세번째 요소로 지정될수 있는 여러가지 허가들이 있는데 읽기허가 r, 쓰기허가 w, 실행허가 x를 포함한다.

기호와 수값방식에 대한 완전히 구체적인 설명은 chmod man페지에서 찾아 볼수 있다.

chown

설명
하나이상의 파일이나 등록부의 사용자와(혹은) 그룹소유권을 변화시킨다.

문법
chown [-Rcfv] [--recursive] [--change] [--quiet] [--verbose] [user] [:.] [group] **파일 ...**

주요기발과 선택항목

- -c/--changes: 소유권이 변화되는 파일들의 이름만을 현시한다.

- -f/--silent/--quiet: 파일의 소유권을 변화시킬수 없을 때 오유통보문의 현시를 금지시킨다.
- -R/--recursive: 변화된 소유권을 가지는 임의의 등록부의 모든 보조등록부에 있는 모든 파일들의 소유권을 반복하여 변화시킨다.
- -v/--verbose: 모든 소유권변화결과를 현시한다.

주의 사용자와 그룹은 이름이나 ID 로 지정될수 있다. 사용자와 그룹은 여러가지 방식으로 결합될수 있다.

- 사용자뒤에 두점이나 점이 있고 그 뒤에 그룹이 있으면 지정한대로 사용자와 그룹의 소유권을 변화시킨다.
- 사용자뒤에 두점이나 점이 있고 그룹이 없으면 지정한대로 사용자의 소유권만을 변화시키며 그룹소유권을 지정된 사용자의 가입그룹으로 변화시킨다.
- 두점이나 점과 그룹이 사용자없이 지정되면 그룹소유권만을 변화시킨다. 이것은 chgrp 와 같다.
- 사용자뒤에 두점이나 점이 없으면 사용자소유권만을 변화시킨다.

clear

설명 화면을 지우고 재촉문과 유표위치를 화면의 첫 행으로 재설정한다.

문법 clear

주요기발과 선택항목 없음

compress

설명 Lempel-Ziv 로 파일이나 표준입구를 압축한다.

문법 compress [-f] [-v] [-c] [-r] [**파일...**]

주요기발과 선택항목

- -c: 파일을 압축할 때 압축된 자료를 표준파일에로가 아니라 표준출구로 출구한다.
- -f: 하드디스크와 련결된 파일의 압축을 강행하는데 이것은 기정에서는 무시된다.
- -r: 반복하여 처리한다. 등록부가 인수로 지정되면 그 등록부와 보조등록부안에 있는 모든 파일들을 압축한다.

- -v: 압축된 매 파일의 축소비률을 현시한다.

주의

파일을 압축할 때 compress 지령은 원래파일을 이름을 그대로 두고 확장자만을 .Z로 바꾼다. 이러한 작용은 -c 기발에 의하여 무시된다. 파일이 지정되지 않으면 표준입구가 압축되며 결과는 표준출구에 나타난다.

cp

설명

파일과 등록부를 복사한다.

문법

cp [-a] [--archive] [-b] [--backup] [-d] [-no-dereference] [-f] [--force] [-i] [--interactive] [l] [--link] [-p] [--preserve] [-r] [--recursive] [-s] [--symbolic-link] [-u] [--update] **목적 원천** cp [**선택항목들**] **원천 ... 목적**

주요기발과 선택항목

- -a/--archive: 원래파일의 기호련결을 유지하고 소유권과 허가들을 보존하면서 파일과 등록부를 반복하여 복사한다. 이것은 -dpR와 같다
- -b/--backup: 원래파일이 덧쓰기되기전에 파일을 여벌복사한다.
- -d/--no-dereference: 련결을 강조하는 곳에서는 파일을 복사하지 않고 련결을 복사한다.
- -f/--force: 덧쓰기가 필요한 현행파일의 제거를 강행한다.
- -i/--interactive: 존재하는 파일을 덧쓰기전에 물어 본다.
- -l/--link: 파일을 복사하지 않고 하드를 련결한다. 이것은 파일에만 적용되고 등록부에는 적용되지 않는다.
- -p/--preserve: 원래파일의 소유권과 허가를 보존한다.
- -R/--recursive: 등록부들을 반복하여 복사한다. 즉 지정된 모든 원천등록부에 대하여 등록부와 그의 보조등록부에 있는 파일들을 복사하며 그동안 대응한 등록부의 구조를 보존한다.
- -s/--symbolic-link: 파일을 복사하지 않고 기호련결을 만든다. 원천파일은 완전한 경로로 지정되여야 한다.

- -u/--upate: 대응하는 원천파일보다 갱신시간이 더 오래된 목적파일들만을 교체한다.

경고

뿌리사용자로 작업할 때는 -f 기발사용에 주의를 돌려야 한다. 뿌리사용자는 일반적으로 모든 파일과 등록부에 대한 쓰기허가를 가지기때문에 덧쓰기되는 중요한 체계파일에서 오유를 발생시킬수 있다.

crontab

설명

사용자의 크론표를 현시하거나 변경시킨다. 크론표는 크론데몬에 의해서 실행되는 일정작성된 동작을 지정한다.

문법

crontab[-u **사용자**] **파일** crontab[-u **사용자**]{-l|-r|-e}

주요기발과 선택항목

- -e: 지령을 실행하거나 사용자나 -u기발로 지정한 사용자의 크론표파일을 편집한다. 요구하는 편집기는 EDITOR 환경변수에 의하여 지정된다.
- -l: 지령을 실행하는 사용자나 -u 기발로 지정한 사용자의 크론표파일내용을 현시한다.
- -r: 지령을 실행하는 사용자나 -u 기발로 지정한 사용자의 크론표파일을 삭제한다.
- -u **사용자**: 지령을 실행하는 사용자가 아니면 어느 사용자의 파일이 작업하는가를 지정한다. 이 기발은 뿌리사용자만이 사용할수 있다.

주의

크론표파일들의 전반적인 형식은 제17장 《일반체계관리》에서 론의된다.

cryptdir

설명

지정된 등록부의 모든 파일들을 암호화한다. 등록부가 지정되지 않으면 현행등록부의 모든 파일들이 암호화된다.

문법

cryptdir [**등록부**]

주요기발과 선택항목

없다

주의

파일을 암호화 할 때 통과암호를 두번 넣어 주어야 한다. 이 통과암호는 파일의 암호를 풀기 위해서 필요한것이다. 암호화된 파일에는 그 이름에 확장자 .crypt 가 붙는다. 파일을 복호할 때 decryptdir 지령을 사용한다.

date

설명

현행체계의 날자와 시간을 현시하거나 설정한다.

문법

date [-u] [--universal] [MMDDhhmm[[CC]YY] [.ss]]

주요기발과 선택항목 | -u/--universal: 그리니치표준시로 시간을 현시한다(또한 만국표준시라고도 한다).

주의 | 날자와 시간이 인수로 주어 지면 지령은 다음과 같이 현시되는 두 자리수로 설정된다.

- *MM*: 월
- *DD*: 일
- *hh*: 시간
- *mm*: 분
- *CC*: 세기 (년의 첫 두 수자)
- *YY*: 년의 마지막 두 수자
- *ss*: 초

뿌리사용자만이 체계시간을 설정할수 있다.

dc

설명 | 사용자가 계산식을 지령재촉문에 넣어 줄수도 있고 지정된 파일의 모든 식들을 처리할수도 있다.

문법 | dc[**파일** ...]

주요기발과 선택항목 | 없다

주의 | dc 에서 사용되는 식의 구체적인 문법은 dc man 페지에서 볼수 있다. 제공된 파일의 식들은 사용자가 처리하려는 추가적인 식들을 넣기 위한 지령대기상태에 들어 가기전에 처리된다.

decryptdir

설명 | 지정된 등록부의 모든 파일의 암호를 해독한다. 등록부가 지정되지 않으면 현재 등록부안의 모든 파일들의 암호가 해독된다. 파일은 encryptdir 지령을 써서 암호화되여야 한다.

문법 | decryptdir [**등록부**]

주요기발과 선택항목 | 없다

주의 | 파일의 암호를 복호하기 위하여 통과암호를 주어야 한다. 파일을 암호화할 때 사용하였던 통과암호를 사용해야 한다. 그렇지 않으면 실패한다.

depmod

설명 | 표준출구에 따르는 모듈을 복귀시킨다. 이것들은 파일에 기억될수 있으며 적재모듈을 설치하는 modprobe 에 의해 사용될수 있다.

문법 depmod 모듈 1.o 모듈 2.o...

주요기발과 선택항목 없다

df

설명 하나이상의 설치된 디스크나 구획들의 빈 공간을 현시한다. 파일(혹은 등록부)이 지정되지 않으면 설치된 모든 파일체계들의 빈 공간이 현시된다. 파일이름이 지정되면 매 파일이 들어 있는 파일체계의 빈 공간이 현시된다.

문법 df [-T] [-t **파일체계형**] [-x **파일체계형**] [--all] [--inodes] [--type=**파일체계형**] [--exclude-type=**파일체계형**] [--print-type] [**파일이름** ...]

주요기발과 선택항목

- -t/--type=**파일체계형**: 지정된 형태의 파일체계에 대한 정보만을 현시한다.
- -T/--print-type: 기록된 매 파일체계의 형을 현시한다.
- -x/--exclude-type=**파일체계형**: 지정된 형태의 파일체계에 대한 정보를 기록하지 않는다.

dir

설명 다른 지정을 하지 않으면 지정된 등록부에 있는 파일들의 목록을 영어자모순서로 현시한다. 다른 등록부가 지정되지 않으면 기정으로 현행등록부의 내용을 현시한다.

문법 dir [-acCGlnrRStuU] [--all] [--no-group] [--numeric-uid-gid] [--reverse] [--recursive] [file ...]

주요기발과 선택항목

- -a/--all: "."으로부터 시작해서 모든 이름항목들을 현시한다.
- -c: 파일을 작성된 시간별로 분류하고 완전한 파일정보를 현시(-l 기발로)할 때 작성된 시간을 현시한다.
- -C: 항목들을 렬로 배렬한다.
- -G/--no-group: 그룹정보를 현시하지 않는다.
- -l: 긴목록형파일들을 현시한다.
- -n/--numeric-uid-gid: 이름대신 사용자 ID와 그룹 ID를 현시한다.
- -r/--reverse: 분류할 때 항목순서를 반대로 한다.

- -R/--recursive : 보조등록부의 차례를 반복하여 현시한다.
- -S : 파일들을 크기에 따라 분류한다.
- -t : 파일을 갱신시간에 따라 분류하고 완전한 파일정보를 현시(-l 기발로)할 때 갱신시간을 현시한다.
- -u : 파일을 마지막호출시간에 따라 분류하고 완전한 파일정보를 현시(-l 기발로)할 때 마지막호출시간을 현시한다.
- -U: 분류하지 않고 등록부순서로 항목들을 현시한다.

dmesg

설명 핵심부고리형완충기를 현시하고 조종한다. 이 완충기에는 많은 bootup 통보문들이 보존되여 있다.

문법 dmesg [-c]

주요기발과 선택항목 -c: 내용들을 현시한후 고리형완충기를 지운다.

dnsdomainname

설명 완전한 자격을 가진 령역이름에 기초한 체계의 DNS 령역이름을 현시한다.

문법 Domainname[-f **파일이름**] [--file **파일이름**]

주요기발과 선택항목 -f/--file **파일이름**: 주콤퓨터이름을 읽기 위한 파일을 지정한다.

dnsquery

설명 지정된 주콤퓨터에 대한 정보들을 알아 내기 위하여 DNS 봉사기에 조회한다.

문법 dnsquery [-n **이름봉사기**] [-t **형태**] [-c **클라스**] [-r **조회**] [-p **재조회주기**] host

주요기발과 선택항목

- -c **클라스**: 탐색되는 레코드의 클라스를 지정한다.
- -n **이름봉사기**: 조회에 사용되는 이름봉사기를 지정한다. 지정하지 않으면 기정이름봉사기가 사용된다.
- -p **재조회주기**: 봉사기가 응답하지 않을 때 얼마나 대기하겠는가를 지정한다.
- -r **조회**: 봉사기가 응답하지 않을 조회를 몇번이나 더 하겠는가를 지정한다.
- -t **형태**: 조회형태를 지정한다.

주의　　　　　　조회형태를 지정할 때 가능한 형태들은 다음과 같다.

- A: 주소만을 알아 본다.
- NS: 주콤퓨터를 위한 이름봉사기를 찾는다.
- CNAME: 주콤퓨터를 위한 표준적인 이름을 찾는다.
- PTR: 령역이름위치를 찾는다.
- SOA: 주콤퓨터를 위한 기록의 시작을 찾는다.
- MX: 령역을 위한 우편교환을 찾는다.
- ANY: 찾을수 있는 모든것을 찾는다(이것은 기정작용이다).

정보클라스를 지정하기 위한 가능한 클라스들은 다음과 같다.

- IN: 인터네트이다(이것은 기정이며 대체로 변화시키지 않는다.).
- HS: Hesiod 이다.
- CHAOS: Chaos 이다.
- ANY: 임의의것이다.

domainname

설명　　　　　　체계의 NIS 령역이름을 현시하거나 설정한다. 인수나 기발이 없으면 기정동작은 현행 NIS 령역이름을 현시한다.

문법　　　　　　domainname [-F **파일**] [--file **파일**] [**이름**]

주요기발과 선택항목　　　-F/--file **파일**: 지령행에서 령역이름이 인수로 요구하는것이 아니라 지정된 파일의 내용에 기초하여 설정된다는것을 지정한다.

du

설명　　　　　　지정된 등록부의 모든 보조등록부는 물론 지정된 모든 파일이나 등록부에 대한 공간사용법을 현시한다.

문법　　　　　　du [-abcksx] [--all] [--bytes] [--total] [--kilobytes] [--summarize] [--one-file-system] [**파일** ...]

주요기발과 선택항목

- -a/--all: 파일과 등록부의 사용정보를 현시한다.
- -b/--bytes: 사용정보를 바이트로 현시한다.
- -c/--total: 모든것에 대한 총 사용형태를 현시한다.

- -k/--kilobytes: 사용정보를 바이트로 현시한다.
- -s/--summarize: 매 인수에 대한 총값을 현시하고 등록부에 있는 매 파일이나 보조등록부에 대한 개별적인 정보를 현시하지 않는다.
- x/--one-file-system: 현재파일체계부분이 아닌 등록부를 뛰여 넘는다.

e2fsck

설명

Linux 의 두번째 확장파일체계상태를 검사한다. 이것은 Linux 구획에 사용된 기정파일체계이다.

문법

e2fsck [-cfnpy][-B **블로크크기**] **장치**

주요기발과 선택항목

- -B blocksize: 초블로크를 탐색할 때 사용하는 지정된 블로크의 크기를 현시한다. 기정으로 프로그람은 초블로크를 찾을 때까지 여러가지 각이한 블로크크기들을 탐색한다.
- -c: badblocks 프로그람으로 나쁜 블로크들을 타색하여 현시한다.
- -f: 표면상 지워 진것으로 보이는 파일체계에 대한 검사를 강행한다.
- -n: 읽기전용상태에서 파일체계를 열고 작용을 위한 모든 질문에 《아니》라고 대답한다.
- -p: 질문이 없이 자동수정을 강행한다.
- -y: 모든 질문에 《예》로 대답한다.

주의

검사되는 장치는 /dev/hda1 혹은 /dev/sdb3 과 같은 완전한 Linux 장치경로를 사용하여 지정되여야 한다. 파일체계를 설치하지 않는것이 좋다. 만일 뿌리파일체계나 파일체계가 설치되였는가를 검사하려면 단일사용자방식으로 되여야 한다.

echo

설명

새 행을 끌고 다니지 않고 본문의 한행을 현시한다(새 행은 기정에 들어 있다.).

문법

echo [-ne] [**문자렬 ...**]

주요기발과 선택항목

- -e: 문자렬의 거꿀빗선뒤에 있는 특수기호를 해석한다.
- -n: 새 행을 출구하지 않는다.

주의

거꿀빗선뒤에 있는 특수기호들은 다음과 같다.

- \ b:공백
- \ f:폼바꾸기
- \ n:새 행
- \ r:돌아가기(행바꾸기)
- \ t:수평꼬리표
- \ \ :거꿀빗선

egrep

설명

지정된 패턴과 일치하는 행들에 대한 파일을 조사한다. 패턴은 확장된 정규식으로 해석된다.

문법

egerp [-bCciLlnvwx] [-**번호**] [-e **패턴**] [-f file] [--byte offset] [--context] [--count] [--regexp=**패턴**] [--file=**파일**] [--ignore=**경우**] [--file-without match] [--file-with-match] [--line-number] [--revert-match] [--ward-regexp] [--line-regexp] **[패턴] 파일 [파일 ...]**

주요기발과 선택항목

- **-번호**: 지정된 문장의 머리부와 꼬리부번호와 일치하는 행을 현시한다.
- -b/--byte-offset: 매 행앞에 일치하는 바이트시작을 인쇄한다.
- -c/--count: 일치하는 행들을 현시하지 않고 식을 일치시키는 총 행수를 간단히 출구한다(-v 와 결합될 때 일치하지 않은 행을 현시한다).
- -C/--context: 문장의 머리부와 꼬리부의 두개의 행과 일치하는 행을 현시한다(이것은 -2와 같다.).
- -e pattern/--regexp=**패턴**: 인수로 제공하지 않고 지정된 정규식을 사용한다.
- -f file/--file=**파일**: 인수로 제공된것이 아니라 지정된 파일에서 볼수 있는 정규식을 사용한다.
- -i/--ignore=case: 패턴과 파일이 둘다 조사되는 경우를 무시한다.
- -l/--files-with-matches: 일치되는 매 행을 현시하지 않고 정규식과 일치되는것을 적어도 하나 포함하는 파일이름을 현시한다.

- -L/--files-without-match: 일치되는 매 행을 현시하지 않고 정규식과 일치하지 않는것을 적어도 하나 포함하는 매 파일의 이름을 현시한다.
- -n/--line-number: 파일에서 매 출구행앞에 자기의 행번호를 붙인다.
- -v/--revert-match: 일치하는 행이 아니라 일치하지 않는 행을 현시한다.
- -w/--word-regexp: 완전한 단어들인 정규식에 일치하는 행들만을 현시한다.
- -x/--line-regexp: 완전한 행들인 정규식에 일치하는 행들만을 현시한다.

false

설명

실패하여 아무것도 하지 못하고 탈퇴상태로 돌아 간다.

문법

false

주요기발과 선택항목

없음

fdisk

설명

구획표를 관리하는 도구들을 제공한다. 기정으로 다른 장치가 인수로 지정되지 않으면 fdisk 는 현행장치로 작업을 시작한다.

문법

fdisk [-l] [-s **구획**] [**장치**]

주요기발과 선택항목

- -l: /dev/hda, /dev/hdb, /dev/sda 로부터 /dev/sdh 까지의 구획표들의 목록을 현시하고 끝낸다.
- -s **구획**: 표준출구에 지정된 구획의 크기를 복귀시키고 끝낸다.

fgrep

설명

지정된 패턴과 일치하는 행들에 대한 파일들을 탐색하고 그 행들을 현시한다. 패턴은 정규식과 대립될 때 고정된 문자렬의 목록으로 해석된다. 문자렬은 새 행에 의해 분리되며 어느 문자렬이나 일치될수 있다.

문법

fgrep [-bCciLlnvwx] [- **번호**] [-e **패턴**] [-f file] [--byte-offset] [--context] [---count] [--regexp= **패턴**] [--file= **파일**] [--ignore= **경우**] [--files-without-match] [--files-with-match] [--line-number] [--revert-match] [--word-regexp]

[--line-regexp] [**패턴**] **파일** [**파일** …]

주요기발과 선택항목

- **-번호**: 문장의 머리부와 꼬리부의 지정된 번호와 일치하는 행들을 현시한다.
- -b/--byte-offset: 매 행의 앞에 일치되는 문자렬의 바이트편위값을 현시한다.
- -c/--count: 일치하는 행들을 현시하지 않고 간단히 식과 일치하는 총 행수를 현시한다(-v 와 결합되면 일치하지 않는 총 행수를 현시한다.).
- -C/--context: 문장의 머리부와 꼬리부의 두 행과 일치하는 행들을 현시한다(이것은 -2 와 같다.).
- -e pattern/--regexp=**패턴**: 인수로 제공된 패턴이 아니라 배렬된 패턴을 사용한다.
- - f file/--file=**파일**: 인수로 제공된 패턴이 아니라 지정된 파일에서 찾은 패턴을 사용한다.
- -i/--ignore=**경우**: 패턴과 파일이 둘 다 탐색되고 있는 경우를 무시한다.
- -l/--files-with-matchs: 일치되는 매 행을 현시하지 않고 간단히 지정된 패턴과 하나라도 일치하는것들을 가지는 매 파일의 이름을 현시한다.
- -L/--files-without-match: 일치되는 매 행을 현시하지 않고 지정된 패턴과 일치하지 않는것들을 가지는 매 파일의 이름을 현시한다.
- -n/--line-number: 출구하는 행앞에 파일의 행번호를 붙인다.
- -v/--revert-match: 일치하는 행이 아니라 일치하지 않는 행을 현시한다.
- -w/--word-regexp: 완전한 단어를 이루는 패턴에 일치하는 행들만을 현시한다.
- -x/--line-regexp: 완전한 행을 이루는 패턴에 일치하는 행들만을 현시한다.

file

설명 파일들의 형을 결정하고 현시한다.

문법 file [-zl] [-f **파일**] **파일** …

주요기발과 선택항목

- **-f 파일:** 지정된 파일로부터 검사되는 파일들의 목록을 읽는다. 이것들은 인수로 제공되는 파일들이 검사되기 전에 검사된다.
- **-L**: 뒤에 놓이는것들은 기호로 련결한다.
- **-z**: 압축된 파일안에 있는 파일들의 형을 볼수 있게 한다.

find

설명

지정된 경로에서 지령행선택항목들이 지정한 모든 기준들과 일치하는 파일들을 찾고 이러한 선택항목들이 지정한 동작을 한다. 경로가 지정되지 않으면 현행등록부에 대한 탐색을 진행한다.

문법

find [**경로** …] [**선택항목**]

주요기발과 선택항목

- **-amin 분**: 지정한 분전에 마지막으로 호출된 파일들을 찾는다.
- **-anewer 파일**: 지정된 파일이 수정된 때보다 더 최근에 호출된 파일들을 찾는다.
- **-atime 날자들**: 24 시간전에 마지막으로 호출된 파일들을 찾는다.
- **-cmin 분**: 지정된 분전에 상태가 마지막으로 변화된 파일들을 찾는다.
- **-cnewer 파일**:상태가 지정된 파일이 수정된 때보다 더 최근에 마지막으로 변화된 파일들을 찾는다.
- **-ctime 날자**: 24 시간내의 지정된 시간에 상태가 변화된 파일들을 찾는다.
- **-empty**: 빈 파일이나 등록부를 찾는다.
- **-exec 지령\ ;**: 지정된 지령을 실행한다. 문자렬{}은 현재 찾은 파일이름으로 바꾸어 지며 지령은 파일들을 찾기 위하여 반복실행된다.
- **-gid 그룹식별자**: 지정된 GID 로 파일을 찾는다.
- **-group 그룹**: 명명된 그룹에 속해 있는 파일들을 찾는다.
- **-ilname 패턴**: 이름이 case-insensitive manner 에서 지적한 패턴과 일치하는 기호련결들을 찾는다.

- **-iname 패턴**: 이름이 case-insensitive manner에서 지정한 패턴과 일치하는 파일들을 찾는다.
- **-ipath 패턴**: 경로가 case-insensitive manner에서 지정한 패턴과 일치하는 파일들을 찾는다.
- **-lname 패턴**: 이름이 case-insensitive manner에서 지정한 패턴과 일치하는 파일들을 찾는다.
- **-maxdepth 준위**: 지정된 경로에서 지정된 준위의 수를 최대로 줄인다.
- **-mindepth 준위**:지정된 경로에서 지정된 준위의 수를 최소로 줄인다.
- **-mmin 분**: 지정된 분전에 마지막으로 갱신된 파일들을 찾는다.
- **-mount**: 현행파일체계와 다른 파일체계에 있는 임의의 등록부를 줄이지 않는다.
- **-mtime 날자들**: 24시간전에 마지막으로 수정된 파일들을 찾는다.
- **-name 패턴**: 이름이 case-insensitive manner에서 지적된 패턴과 일치하는 파일들을 찾는다.
- **-newer 파일**: 지정된 파일보다 더 최근에 수정된 파일들을 찾는다.
- **-nogroup**: 수값 GID가 현행그룹과 일치하지 않는 파일들을 찾는다.
- **-nouser**: 수값 UID가 현행사용자와 일치하지 않는 파일들을 찾는다.
- **-ok 지령\ ;**: 사용자를 재촉한후에 찾은 매 파일들에 대하여 지정된 지령을 실행한다. 문자렬 {}는 현재 바뀌여 진 파일이름으로 바꾸어 진다.
- **-path**: 경로가 case-insensitive manner 에서 지정한 패턴과 일치하는 파일들을 찾는다.
- **-perm 방식**: 권한이 지정된 방식과 완전히 일치하는 파일들을 찾는다. +방식이 사용되면 지정된 권한의 어느 비트든지와 일치하며 -방식이 사용되면 지정된 권한의 모든 비트들과 일치한다.
- **-print**: 찾은 모든 파일들의 완전한 파일이름을 현시한다.
- **-regex 패턴**: 이름이 지정된 정규식과 일치하는 파일을 찾는다.
- **-size 크기 [bckw]**: 지정된 묶음에서 지정된 크기의 파일들을 찾는다. 묶음은 b(512byte 단위블로크), c(byte 단위), k(kbyte 단위), w(2byte word 단위)

를 포함한다.

- **-type 형**: 지정된 형과 일치하는 파일들을 찾는다. 형에는 d(등록부), f(정규파일), l(기호련결)이 있다.
- **-uid 사용자 ID**: 지정된 UID를 가진 파일을 찾는다.
- **-user 사용자이름**: 지정된 사용자이름이나 UID가 인정하는 파일들을 찾는다.

주의　　분이나 날자와 같은 수값시간정보를 제공할 때 기정으로 정확히 일치되여야 한다. 수앞에 +가 있으면 지정된 수보다 더 큰 값과 일치하며 -가 있으면 지정된 수보다 더 작은 값과 일치한다.

finger

설명　　국부체계나 원격체계에서 사용자에 대한 정보를 찾는다. 국부체계에서 사용자들은 자기의 사용자이름이나 첫번째 또는 마지막이름으로 지정되며 원격체계에서는 user name@host로 지정된다. 국부체계에서 사용자들이 지정되지 않으면 현행체계에 등록된 모든 사용자들이 현시된다. 사용자이름이 없는 주콤퓨터가 @host형태로 지정되면 원격체계에 등록된 모든 사용자들의 목록이 현시된다.

문법　　finger [**사용자** …]

주요기발과 선택항목　　없음

free

설명　　기억기의 빈 용량과 사용된 용량을 현시한다.

문법　　free [-b|-k|-m] [-s **지연**] [-t]

주요기발과 선택항목

- -b: 기억기의 총 용량을 byte단위로 현시한다.
- -k: 기억기의 총 용량을 kbyte단위로 현시한다.
- -m: 기억기의 총 용량을 mbyte단위로 현시한다.
- **-s 지연**: 초단위로 지정된 지연시간간격으로 계속되는 보고를 현시한다.
- -t: 총 여분행을 현시한다.

gpasswd

설명　　/etc/group 파일을 관리한다. 기발이 없으면 gpasswd는 지정된 그룹의 통과암호를 변경시킨다.

문법　　gpasswd **그룹**

gpasswd -a **사용자 그룹**
gpasswd -d **사용자 그룹**
gpasswd -R **그룹**
gpasswd -r **그룹**
gpasswd [-M **사용자,…**] **그룹**

주요기발과 선택항목

- -a **사용자**: 그룹에 사용자를 추가한다.
- -d **사용자**: 그룹으로부터 사용자를 제거한다.
- -M **사용자,…**: 그룹의 성원인 하나이상의 사용자를 지정한다.
- -r: 그룹통과암호를 지운다.
- -R: newgrp지령으로 만든 그룹에 대한 호출을 금지한다.

grep

설명

지정된 패턴과 일치하는 행들에 대한 파일을 탐색하고 그 행들을 현시한다.

문법

grep [-bCcEFGiLlnvwx] [-number] [-e] [-f] [--basic-regexp] [--byte-offset] [--extended-regexp [--fixed-strings] [--byte-offset] [--context] [--count] [--regexp=] [--file= **파일**] [--ignore= **경우**] [--files-without-match] [--files-with-match] [--line-number] [--revert-match] [--word-regexp] [--line-regexp] [**패턴**] **파일** [**파일** …]

주요기발과 선택항목

- -number: 문맥의 머리부와 꼬리부의 지정된 번호와 일치하는 행들을 현시한다.
- -b/--byte-offset: 매 행의 앞에 일치되는 문자렬의 바이트편위값을 현시한다.
- -c/--count: 일치하는 행들을 현시하지 않고 간단히 식과 일치하는 총 행수를 현시한다(-v 와 결합되면 일치하지 않는 총 행수를 현시한다.).
- -C/--context: 문맥의 머리부와 꼬리부의 두 행과 일치하는 행들을 현시한다(이것은 -2 와 같다.).
- -e **패턴**/--regexp=**패턴**: 인수로 지정된것이 아니라 지정된 패턴을 사용한다.
- -E/--extented-regexp: 확장된 정규식으로 패턴을 처

리한다(**egrep** 와 류사하다.).

- **-f file/--file=파일**: 인수로 지정된 패턴이 아니라 지정된 파일에서 찾은 패턴을 사용한다.
- **-F/--fixed-strings**: 패턴을 문자렬(문자렬중의 어느 하나라도 일치해야 한다.)로 분리된 새 행의 목록으로 취급한다. 이것은 fgrep 와 같다.
- **-G/--basic-regexp**: 패턴을 기본정규식으로 처리한다.
- **-i/--ignore=경우**: 패턴과 파일이 둘다 조사되고 있는 경우를 무시한다.
- **-l/--files-with-matchs**: 일치되는 매 행을 현시하지 않고 간단하게 지정된 패턴과 하나라도 일치하는것을 가지는 매 파일의 이름을 현시한다.
- **-L/--files-without-match**: 일치되는 매 행을 현시하지 않고 지정된 패턴과 일치하지 않는것을 가지는 매 파일의 이름을 현시한다.
- **-n/--line-number**: 파일에서 출구되는 모든 행들의 앞에 행번호를 붙인다.
- **-v/--revert-match**: 일치하는 행이 아니라 일치하지 않는 행을 현시한다.
- **-w/--word-regexp**: 완전한 단어를 이루는 패턴에 일치하는 행들만을 현시한다.
- **-x/--line-regexp**: 완전한 행을 이루는 패턴에 일치하는 행들만을 현시한다.

주의

grep 로 사용되는 정규식의 문법은 **grep** man 페지를 본다.

groupadd

설명

새로운 그룹을 만든다.

문법

groupadd [-g 그룹식별자] [-o] [-r] [-f] 그룹

주요기발과 선택항목

- **-f**: 이미 퇴장한 그룹을 추가하려고 할 때 프로그람이 끝나지 않게 한다. 이 경우에 그룹은 바뀌여 지지 않는다.

- **-g 그룹식별자**: 자동적으로 할당된 값이 아니라 그룹의 GID 를 사용한다.
- -o: 그룹 ID 들이 유일하지 않아도 된다는것을 현시한다.
- -r: 체계회계를 499 보다 작은 그룹 ID 와 합한다.

groupdel

설명　　그룹을 지운다.

문법　　**groupdel 그룹**

주요기발과 선택항목　　없음

groupmode

설명　　현행그룹을 수정한다.

문법　　**groupmode [-g 그룹식별자 [-o]] [-n 그룹이름] 그룹**

주요기발과 선택항목

- **-g 그룹식별자**: 지정된 그룹의 그룹 ID 를 새로운 GID 로 바꾼다. 이 값은 -o 가 지정되지 않는 한 유일해야 한다.
- **-n 그룹이름**: 지정된 그룹의 이름을 새로운 그룹이름으로 바꾼다.
- -o: 그룹 ID 가 유일하지 않아도 된다는것을 현시한다.

groups

설명　　하나이상의 사용자들이 속한 그룹을 현시한다. 사용자가 지정되지 않으면 지령을 실행하는 사용자가 속하는 그룹을 현시한다.

문법　　**group [사용자이름 …]**

주요기발과 선택항목　　없음

grpck

설명　　/etc/group 나 /etc/gshadow 와 같은 그룹파일의 초기상태를 검사한다. 파일이 지정되지 않으면 기정파일들이 검사된다.

문법　　**grpck [-r] [그룹그림자]**

주요기발과 선택항목　　-r: 파일에 어떤 변화도 생기지 않게 하면서 읽기방식으로만 처리한다.

gunzip

설명

gzip지령(compress지령과 zip지령과 같은)에 의해 압축된 파일을 푼다.

문법

gunzip [-cflrt] [--stdout] [--to-stdout] [--force] [--list] [--recursive] [--test] [이름 …]

주요기발과 선택항목

- -c/--stdout/--to-stdout: 초기파일이 변화되지 않게 하면서 기정출구에로 출구한다. 기정에 의해 gunzip는 초기에 압축된 파일을 압축되지 않은 파일의 판본으로 바꾼다.
- -f/--force: 대응하는 파일이 이미 존재하고 있는 경우에 조차도 풀기를 강행하며 풀린 파일들로 덧쓰기한다.
- -l/--list:압축을 풀지 않은 상태에서 압축파일에 있는 파일들을 현시한다.
- -r/--recursive: 등록부구조를 따라 내려 가면서 지령행에 인수로 명명된 등록부의 보조등록부에 있는 모든 파일을 반복하여 푼다.
- -t/--test: 압축된 파일들의 완전성을 검사한다.

gzexe

설명

실행가능한 압축파일을 만든다. 2진파일이나 스크립트를 gzexe로 압축한다면 압축되지 않았을 때처럼 그것을 실행시킬수 있다. 기억기에서 간단히 풀리여 실행되며 하드디스크에는 압축된 판본이 남아 있다.

문법

gzexe [-d] [이름 …]

주요기발과 선택항목

-d: 지정된 파일을 압축하는것이 아니라 푼다.

주의

filename으로 명명된 파일을 압축하면 초기에 압축되지 않은 파일은 filename~으로 복사되며 압축된 파일은 filename이름을 기억한다. 만일 그렇게 동작하는가를 보기 위하여 압축된 실행파일을 일단 시험하면 압축되지 않은 복사물을 지울수 있다.

gzip

설명

Lempel-ziv 부호화를 써서 파일들을 압축한다. 일반적으로 결과파일은 초기압축되지 않은 파일로 교체되며 .gz확장자를 가진다.

문법

gzip [-cdflrt] [--decompress] [--uncompress] [--stdout]

[--to-stdout] [--force] [--list] [--recursive] [-test] [이름 …]

주요기발과 선택항목

- -c/--stdout/--to-stdout: 초기파일이 변화되지 않게 하면서 표준출구에로 출구한다. 기정으로 gzip는 초기압축되지 않은 파일들을 압축된 파일들의 판본으로 바꾼다.
- -d/--decompress/--uncompress: 지정된 파일들을 압축하는것이 아니라 gunzip와 같이 그것들을 푼다.
- -f/--force: 대응하는 파일이 이미 존재하는 경우에도 압축을 강행하고 압축된 파일로 덧쓰기한다.
- -l/--list: 압축된 파일들의 목록을 현시한다.
- -r/--recursive: 등록부구조를 따라 가면서 지령행에 인수로 명명된 등록부의 보조등록부에 있는 모든 파일을 반복하여 압축한다.
- -t/--test: 압축된 파일의 초기상태를 검사한다.

halt

설명

체계를 정지시킨다. 체계가 실행준위 0이나 6에 없으면 shutdown 프로그람을 호출하여 실현한다.

문법

halt [-n] [-d] [-f] [-i]

주요기발과 선택항목

- -d: /var/log/wtmp에 체계의 정지상태를 기입하지 않는다. 기정으로 체계의 정지상태는 이 파일에 기록된다.
- -f: shutdown 프로그람을 호출하지 않고 체계의 정지나 재기동을 강행한다.
- -i: 체계가 정지되기전에 망대면부를 닫는다.
- -n: 체계가 정지되기전에는 파일체계들을 동기시키지 않는다.
- -w: 체계의 정지상태에 대한 기록을 /var/log/wtmp에 쓰지만 실제로는 체계를 정지시키지 않는다.

경고

이 지령을 사용할 때 주의해야 한다. 디스크들을 동기시키지 않고 체계를 정지시키는 -n기발은 그것을 해제하기전에 파일체계를 동기시키는것이 실패하면 디스크에 기억된 자료를 파괴할수 있기때문에 특별히 주의해서 써야 한다.

head

설명

하나이상의 파일들의 머리부분을 현시한다. 기정으로 다른 것이 지정되지 않는 한 매 파일의 첫 10개 행을 현시한다. 파일이름이 제공되지 않으면 파일에 관한 규칙에 따라 표준입력으로부터 자료를 읽고 자료의 첫 부분을 현시한다.

문법

head [-c **개수**[bkm]] [-n **개수**] [-qv] [--bytes **개수**[bkm]] [--lines **개수**] [--quiet] [--silent] [**파일** …]

주요기발과 선택항목

- -c/--bytes **개수**: 매 파일들의 시작으로부터 지정된 바이트수를 현시한다. b는 512byte블로크를, k는 kbyte를, m은 Mbyte를 현시한다.
- -n/--lines **개수**: 매 파일의 시작으로부터 지정된 행수를 현시한다.
- -q/--quiet/--silent: 여러개의 파일들이 처리되고 있을 때 파일이름머리부들에 대한 현시를 금지한다.

hostname

설명

체계의 주콤퓨터이름을 현시하거나 설정한다. 기발이나 인수가 주어 지지 않으면 체계의 주콤퓨터이름이 현시된다.

문법

hostname [-a] [--alias] [-d] [--domain] [-f] [--fqdn] [-i] [--ip-address] [--long] [-s] [--short] [-y] [--yp] [--nis]

주요기발과 선택항목

- -a/--alias: 가능하다면 주콤퓨터의 별명이름을 현시한다.
- -d/--domain: 주콤퓨터의 DNS령역이름을 현시한다.
- -f/--fqdn/--long: 완전히 권한을 가진 주콤퓨터의 령역이름을 현시한다.
- -i/--ip-address: 주콤퓨터의 IP주소를 현시한다.
- -s/--short: 령역이름이 없이 주콤퓨터이름을 현시한다.
- -y/--yp/--nis: 체계의 NIS령역이름을 현시한다.

id

설명

지정된 사용자를 위한 실제적이고 유용한 사용자정보와 ID정보를 현시한다. 사용자가 지정되지 않으면 id를 실행하고

있는 사용자의 정보를 현시한다.

문법

id [-gnruG] [--group] [--name] [--real] [--user] [--groups] [**사용자이름**]

주요기발과 선택항목

- -g/--group: 그룹 ID 만을 현시한다.
- -G/--groups: 보조적인 그룹들만을 현시한다.
- -n/--name: ID 번호대신에 사용자나 그룹이름을 현시한다. -u, -g 나 -G 와 결합해서 리용된다.
- -r/--real: 리용가능한 사용자대신에 실제적인 사용자나 그룹 ID 를 현시한다. -u, -g 나 -G 와 결합해서 리용된다.
- -u/--user: 사용자 ID 만을 현시한다.

ifconfig

설명

망대면부를 구성하거나 선택항목들이 제공되지 않으면 그의 상태를 현시한다. 인수들이 제공되지 않으면 모든 대면부의 현행상태를 현시한다.

문법

ifconfig **대면부 선택항목 주소**

주요기발과 선택항목

- **대면부**: 망대면부의 이름(례를 들어 eth0 이나 eth1)을 지정한다.
- up: 지정된 대면부를 기동시킨다.
- down: 지정된 대면부를 해제한다.
- netmask **주소**: 대면부를 위한 망마스크를 설정한다.
- Broadcast **주소**: 대면부를 위한 방송망주소를 설정한다.
- pointtopoint **주소**: 두 콤퓨터사이의 직접련결을 나타내는 대면부들에 대하여 점대점방식을 쓸수 있다. 또한 다른 련결끝주소를 설정한다.
- **주소**: 대면부를 위한 주콤퓨터이름이나 IP 주소를 지정한다. 이것은 필요하다.

ifdown

설명

eth0 이나 eth1 과 같은 지정된 대면부를 무시한다.

문법 **ifdown 대면부**

주요기발과 선택항목 없음

ifport

설명 지정된 망대면부의 송수신형을 설정한다.

문법 **ifport 대면부 형**

주요기발과 선택항목 **형**: 송수신형을 지정한다. 가능한 형은 다음과 같다.

auto(자동선택), **10baseT**(꼬임쌍선 Ethenet), **10base2**(동축케블 Ethenet), **aui**(AUI 대면부 Ethenet), **10baseT**(꼬임쌍선 Past Ethenet)

ifup

설명 eth0이나 eth1과 같은 지정된 대면부를 사용할수 있다.

문법 **ifup 대면부**

주요기발과 선택항목 없음

insmod

설명 현재 핵심부에 적재가능모듈을 설치한다.

문법 insmode [-fpsxX] [-o **모듈_이름**] **대상_파일** [symbol=**값** …]

주요기발과 선택항목

- -f: 핵심부판본과 기대된 핵심부판본이 일치하지 않을 때에도 모듈을 적재할수 있다.
- -o **모듈**: 모듈을 위한 대상파일에 있는 이름을 리용하지 않고 모듈의 이름을 명백하게 짓는다.
- -p: 그것이 정확히 적재되게 하는 모듈을 조사한다.
- -s: 표준출구가 아니라 체계가입데몬에 능동적으로 가입한다.
- -x: 모듈의 외부기호들을 조사하지 않는다.
- -X: 모듈의 외부기호들을 조사한다(이것은 기정이다.).

kbd_mode

설명 건반방식을 현시하거나 설정한다.

문법 kbd_mode [-a|-u|-k|-s]

주요기발과 선택항목

- -a: ASCII(XLATE)방식으로 건반을 설정한다.
- -K: 건반코드(MEDIUMRAW)방식으로 건반을 설정한다.
- -s: 주사(RAW)방식으로 건반을 설정한다.
- -u: UTF-8(UNICODE)방식으로 건반을 설정한다.

kbdrate

설명

건반의 반복률과 지연시간을 설정한다.

문법

kbdrate [-r 속도] [-d 미리초]

주요기발과 선택항목

- **-d 미리초**: 지정된 미리초로 지연(반복하기전에)을 설정한다.
- **-r 초당 문자수**: 지정된 초당 문자수로 반복속도를 설정한다. 모든 값들이 다 가능한것이 아니라 다음의 값들로부터 선택하여야 한다.

  2.0, 2.1, 2.3, 2.5, 2.7, 3.0, 3.3, 3.7, 4.0, 4.3, 4.6, 5.0, 5.5, 6.0, 6.7, 7.5, 8.0, 8.6, 9.2, 10.0, 10.9, 12.0, 13.3, 15.0, 16.0, 17.1, 18.5, 20.0, 21.8, 24.0, 26.7, 30.0

kill

설명

하나이상의 실행중에 있는 프로쎄스들에 닫기신호를 보낸다.

문법

kill [-s 신호|-p] 프로쎄스식별자 …

kill -l

주요기발과 선택항목

- -l: 신호이름의 목록을 현시한다.
- -p: 그것을 신호로 보내지 않고 지정된 프로쎄스의 프로쎄스 ID를 현시한다.
- **-s 신호**: 지정된 신호를 지정된 프로쎄스에 보낸다.
- **프로쎄스식별자**: 프로쎄스 ID 혹은 그것의 이름을 지정한다. 프로쎄스를 이름으로 지정하면 지정된 이름을 가진 모든 프로쎄스들은 신호를 접수한다.

killall

설명
공동프로쎄스이름을 공유하는 모든 프로쎄스에 신호를 보낸다.

문법
killall [-eiw] [-신호] 프로쎄스 …
killall -l

주요기발과 선택항목

- -e: 프로그람이 15 문자보다 더 긴 프로쎄스이름들에서 정확히 일치하는것들에만 신호를 보내도록 한다.
- -i: 매 프로쎄스에 신호를 보내기전에 확인상태를 물어 본다.
- -l: 신호이름목록을 현시한다.
- -w: 닫으려는 프로쎄스들이 모두 닫길 때까지 기다린다. 이것은 killall 이 프로쎄스를 닫는것이 실패하면 무의미하게 기다릴수 있다는것을 의미한다.

ksyms

설명
주소이름과 정의된 모듈을 포함하여 제출되는 핵심부기호들에 대한 정보를 현시한다.

문법
ksyms [-a] [-m]

주요기발과 선택항목

- -a: 실지 핵심부로부터 그것들을 포함하는 모든 기호들을 현시한다.
- -m: 모듈의 주소와 크기를 포함하는 모듈정보를 현시한다.

last

설명
/var/log/wtmp 의 내용에 기초한 사용자망가입과 사용자망차단에 대한 리력을 현시한다. tty0 이나 tty1 과 같은 tty 가 지정되면 tty 가 현시되는데만 망접속한다.

문법
last [-R] [-number] [-n 번호] [-adx] [이름 …] [tty …]

주요기발과 선택항목

- -a: 마지막항목에 현시될 주콤퓨터이름을 강행한다.
- -d: 원격망접속인 경우에 주콤퓨터이름으로 모든 IP 주소들을 현시한다.
- -n 개수/-number: 현시하려는 리력의 행수를 지정한다.
- -R: 보고서에 주콤퓨터이름의 현시를 금지한다.
- -x: 체계를 정지시키거나 망접속과 망차단에 따라 현시되는 실행준위를 변화시킨다.

ldd

설명 한개이상의 프로그람에 대한 공유서고종속성을 현시한다.

문법 ldd [-dr] **프로그람** …

주요기발과 선택항목

- -d: 재배치를 수행한후 빠진 기능들을 보고한다.
- -r: 재배치를 수행한후 빠진 자료객체들과 기능들을 보고한다.

less

설명 반대방향으로 탐색하면서 한번에 한 화면씩 본문파일을 현시한다.

문법 less [-aeEGiINrsS] **파일** …

주요기발과 선택항목

- -a: 화면의 마지막행으로부터 탐색을 시작한다. 기정으로 탐색은 현재 보이는 본문에 대하여 한다.
- -e: 파일의 끝을 두번째로 만나면 less를 끝낸다. 그렇치 않으면 "q"지령으로 끝낸다.
- -E: 파일의 끝을 첫번째로 만나면 less를 끝낸다.
- -G: 조사과정에 찾은 문자렬들에 대한 강조를 금지한다.
- -i: 탐색을 반응이 없는 상태로 되게 한다. 이것은 탐색패턴이 대문자들을 포함하면 그것은 무시된다.
- -I: 탐색패턴이 대문자들을 포함해도 탐색을 반응이 없는 상태로 되게 한다.
- -N: 매 행의 시작에 행번호가 현시되게 한다.
- -r: 카레트표기법을 리용하여 가공되지 않은 조종문자들이 현시되게 한다(실례로 Ctrl+A는 ^A이다.).
- -s:련속적인 공백행들을 하나의 공백행으로 압축한다.
- -S: 화면의 폭보다 더 넓은 행들을 다음행에 라렬하지 않고 자른다.

lilo

설명 Linux 시동적재기를 설치한다.

문법 lilo [-C **파일**] [-d **데시초**] [-q] [-D **표식**] [-u **장치**]

주요기발과 선택항목

- -C **파일**: 시동적재프로그람을 적재하는데 사용하는 특수한 구성파일을 지정한다. 기정구성파일은 /etc/lilo.conf이다.
- -d **데시초**: 기정핵심부을 적재하기전에 시동하는 동안 LILO재촉문에서 기다리는 시간을 데시초로 현시한다.
- -D **표식**: 구성파일에 있는 첫 핵심부가 아니라 기정핵심부로 지정된 표식을 가진 핵심부를 사용한다.
- -q: 기동되는 핵심부들을 작성하면서 현재 현시된 파일들을 현시한다.
- -u **장치**: 지정된 장치에 대한 시동적재기를 해체한다.

listalias

설명

사용자와 체계의 별명들을 현시한다. 정규식이 있으면 그 식에 맞는 별명들만을 현시한다.

문법

listalias [-s|-u] [**정규식**]

주요기발과 선택항목

- -s: 체계별명들만 현시한다.
- -u: 사용자별명들만 현시한다.

ln

설명

파일들사이에 련결을 만든다. 마지막인수가 등록부이면 지정된 서로 다른 원천파일들은 지정된 등록부안에서 같은 이름을 가진 파일에 련결된다.

문법

ln [--bis] [--backup] [--interactive] [--simbolic] **원천** [**목적**]

ln [--bis] [--backup] [--interactive] [--simbolic] **원천** … **등록부**

주요기발과 선택항목

- -b/--backup: 삭제되는 파일의 여벌복사물을 만든다.
- -i/--interactive: 목적파일을 삭제할 필요가 있을 때 입력한다.
- -s/--symbolic: 하드련결대신에 기호련결을 만든다.

loadkeys

설명

한개이상의 지정된 파일들로부터 구성건변환표를 적재한다. 파일들이 지정되지 않으면 표준입구로부터 정보를 읽는다.

문법

loadkeys [-c --clearcompose] [-d --default] [-m

--mktable] [-s --clearstrings] [**파일** …]

주요기발과 선택항목

- -c/--clearcompose: 새로운 항목들을 적재하기전에 현재의 악센트표를 지운다. 항목들이 지정되지 않으면 표는 비여 있다.
- -d/--default: 기정건넘기기를 적재한다.
- -m/--mktable: 현재 넘기기의 표준출구에 대한 표를 현시한다.
- -s/--clearstring: 핵심부문자렬표를 지운다.

logger

설명

체계로그안에 항목들을 배치한다. 통보문이 지정되지 않고 입구파일이 제공되지 않으면 표준입구가 체계로그에 기입된다.

문법

logger [-is] [-f **파일**] [-p **우선순위**] [-t **꼬리표**] [**통보문**…]

주요기발과 선택항목

- -f **파일**: 체계로그에 지정된 파일을 기록한다.
- -i: 로그파일의 매 행에 대한 입구점을 만드는 프로세스 ID를 배치한다.
- -p **우선순위**: 로그항목의 우선순위를 나타낸다.
- -s: 체계로그외에 표준오유에 대한 통보문을 기록한다.
- -t **꼬리표**: 지정된 꼬리표를 가진 로그항목의 매행을 현시한다.

login

설명

체계에 가입한다.

문법

login **사용자이름**

주요기발과 선택항목

없음

logname

설명

사용자이름을 현시한다.

문법

logname

주요기발과 선택항목

없음

logrotate

설명

현재파일을 우편으로 보내고 보관을 위하여 압축하면서 등

록파일을 회전시킨다.

문법

logrotate [-s|--state **파일**] **구성파일**

주요기발과 선택항목

- -s/--state **파일**: 기정 /var/lib/logrotate.status 대신에 지정된 상태파일을 리용한다.

lpd

설명

붙어 있는 인쇄기와 원격인쇄기들에 대한 현시를 조종하는 행인쇄기완충기데몬을 실행한다. 포구번호가 지정되면 기정포구가 요청을 들어 주는것을 무시한다.

문법

lpr [-l] [-**포구**]

주요기발과 선택항목

-l: 유용한 망요구들을 기록한다.

lpq

설명

인쇄기완충기의 현재상태를 시험하고 현시한다. 인쇄기가 지정되지 않으면 기정인쇄기가 의심된다. 보통 의심되는 인쇄기에 대한 모든 일감들은 특수한 일감번호가 주어 지지 않는 한 현시된다. 사용자들이 지정되면 이러한 사용자들의 인쇄일감들이 의심되는 인쇄기들에 대하여 현시된다.

문법

lpq [-l] [-P **인쇄기**] [**일감,**…][**사용자,**…]

주요기발과 선택항목

- -l: 한 행에 꼭 맞지 않아도 일감항목들을 구성하는 파일에 대한 모든 정보를 현시한다.
- -P **인쇄기**: 지정된 인쇄기를 의심한다.

lpr

설명

하나이상의 파일들을 지정된 인쇄기완충기에 현시한다. 파일이 지정되지 않으면 표준입력을 인쇄기완충기에 보낸다. 인쇄기가 지정되지 않으면 자료를 기정인쇄기에 보낸다.

문법

lpr [-P **인쇄기**] [-#**수**] [-C **클라스**] [-J **일감**] [-i [**렬수**]] [-hlmrs] [**파일** …]

주요기발과 선택항목

- -#**수**: 매 파일을 인쇄하기 위한 복사회수를 지정한다. 기정은 한번 복사하는것이다.
- -C **클라스**: 머리부페지에 있는 주콤퓨터이름이 아니라 지정된 클라스이름을 현시한다.
- -h: 머리부페지의 인쇄를 금지한다.
- -i [**렬수**]: 출구가 지정된 공백수로 혹은 개수가 지정되

지 않으면 8개 문자로 발행되여야 한다는것을 나타낸다.

- **-J 일감**: 머리부페지에 있는 파일이름대신에 지정된 일감이름을 현시한다.
- -l: 페지파괴를 금지시키는 한편 조종문자들이 현시되게 한다.
- -m: 인쇄일감이 끝날 때 사용자에게 전자우편을 보낸다.
- **-P 인쇄기**: 지정된 인쇄기로 현시한다.
- -r: 그것들을 인쇄한후에 파일을 삭제한다.
- -s: 완충기등록부에 파일을 복사하는것이 아니라 인쇄되고 있는 파일에 대한 기호련결을 만든다.

lprm

설명 지정된 인쇄렬로부터 하나이상의 일감을 지운다. 만일 인쇄기가 지정되지 않으면 기정인쇄렬로부터 일감을 삭제하려고 한다. 사용자이름이 지정되면 특수한 일감들이 지정되지 않는 한 사용자가 승인하는 모든 일감들을 삭제한다.

문법 lprm [**-P 인쇄기**] [**일감 …**] [**사용자 …**]

주요기발과 선택항목 **-P 인쇄기**: 지정된 인쇄완충대기렬로부터 일감들을 지운다.

ls

설명 파일과 등록부의 목록을 현시한다. 만일 등록부나 파일이 지정되지 않으면 현행등록부의 내용들을 현시한다. 기정으로 내용들은 자모순서대로 분류된다.

문법 ls [-acdlrRsStuX] [--all] [--time=ctime] [--time=status] [--directory] [--format=long] [--format=verbose] [--reverse] [--recursive] [--size] [--sort-size] [--sort=time] [--time=atime] [--time=access] [--time=use] [--sort=extension] [**파일/등록부 …**]

주요기발과 선택항목

- -a/--all: "."으로 시작되는것들을 포함하여 등록부의 모든 파일을 현시한다.
- -c/--time=ctime, --time=status: 파일의 변경시간에 따라 항목들을 분류한다.
- -d/--directory: 등록부의 내용은 보여 주지 않고 등록부의 이름만을 현시한다. 기정으로 등록부내용을 현시한다.

- -l/--format=long/--format=verbose: 파일의 형태, 허가, 소유자들과 크기를 포함한 긴 형태로 파일들을 현시한다.
- -r/--reverse: 반대순서로 파일들을 현시한다.
- -R/--recursive: 모든 보조등록부의 내용을 반복하여 현시한다.
- -s/--size:파일의 크기를 k byte 단위로 현시한다.
- -S/--sort=size: 가장 큰 파일로부터 시작하여 파일의 크기에 따라 항목들을 분류한다.
- -t/--sort=time:제일 새로운 파일로부터 시작하여 시간에 따라 항목들을 분류한다.
- -u/--time=atime/--time=access/--time=usex: 항목들을 마지막호출시간에 따라 분류한다.
- -X/--sort=extension: 파일들을 파일확장자에 따라 자모순서로 분류한다.

lsdev

설명 설치된 하드웨어에 대한 정보를 현시한다.

문법 lsdev

주요기발과 선택항목 없음

lsmod

설명 적재된 모듈들의 목록을 현시한다.

문법 lsmod

주요기발과 선택항목 없음

mail

설명 전자우편을 발송하거나 입수한다. 사용자는 그것이 표준입구를 통하여 제공되지 않는 한 통보문을 재촉하여야 한다. 또한 통보문을 발송할 때 주제와 같은 다른 정보가 기발로 제공될수 있거나 프로그람이 정보를 재촉할수 있다. 만일 선택항목이나 인수가 지정되지 않으면 읽기 위하여 현행사용자의 우편통을 펼친다.

문법 mail [-s **제목**] [-c **주소,…**] [-b **주소,…**] **주소**
mail -f [**우편통**]
mail -u [**사용자**]

주요기발과 선택항목

- **-b 주소,…**:리해할수 없는 출구통보문의 목적인 복사물을 접수하는 주소들의 목록을 지정한다.
- **-c 주소,…**: 출구통보문의 맹목적인 복사물을 접수하는 주소들의 목록을 지정한다.
- **-f [우편통]**: 자기의 우편함이나 지정된 우편함으로부터 우편물을 읽는다.
- **-s 제목**: 출구하는 통보문의 제목행을 지정한다.
- **-u 사용자**: 통보문을 읽기 위하여 지정된 사용자의 내부함을 펼친다.

주의 통보문을 읽기 위한 **mail** 지령을 리용하는 기술은 **mail** man 페지에서 설명한다.

mailq

설명 출구하는 우편대기렬의 내용을 현시한다.

문법 **mailq**

주요기발과 선택항목 없음

mailto

설명 하나이상의 수신자들에게 하나의 전자우편을 발송한다. 지령행에 수신자들이 지정되여 있지 않으면 사용자는 수신자를 재촉한다. 표준출구가 제공되지 않으면 사용자는 통보내용을 보기 위해서 재촉한다.

문법 **mailto [-a 문자모임] [-c 주소,…] [-s 제목] [수신자 …]**

주요기발과 선택항목

- **-a 문자모임**: ISO-8859-8 과 같은 교체문자모임을 지정한다. 기정은 US-ASCII 이다.
- **-c 주소,…**: 맹목적인 복사주소를 지정한다.
- **-s 제목**: 통보문의 제목행을 지정한다. 제목이 한 단어이상이면 인용문안에 그것을 입력해야 한다.

주의 통보문구성을 끌내기 위하여 Ctrl+D 를 리용하거나 공백행우에 **a.** 만을 입력한다.

man

설명 지정된 지령에 대한 안내페지를 현시한다.

문법 **Man 지령**

주요기발과 선택항목 **지령**: 안내페지가 현시되는 지령을 지정한다.

mattrib

설명 DOS 의 플로피디스크와 같은 MS-DOS 파일체계에 있는 파일의 속성을 바꾼다. 이것은 DOS 의 ATTRIB 지령과 류사하다.

문법 mattrib [-a|+a] [-h|+h] [-r|+r] [-s|+s] msdos **파일** [msdos **파일** …]

주요기발과 선택항목

- +a|-a: 문서고비트를 설정 또는 해제한다.
- +h|-h: 숨겨 진 비트를 설정 또는 해제한다.
- +r|-r: 읽기전용비트를 설정 또는 해제한다.
- +s|-s: 체계비트를 설정 또는 해제한다.

주의 비트를 설정하기 위하여 선택항목의 앞에 +를 , 비트를 해제하기 위하여 -를 사용한다.

mbadblocks

설명 나쁜 블로크를 찾기 위하여 DOS의 플로피디스크를 검사한다. 그것들은 디스크의 FAT 에 현시된다.

문법 mbadblocks **구동기:**

주요기발과 선택항목 없음

mcd

설명 플로피디스크와 같은 MS-DOS파일체계우에서 등록부를 바꾼다. 인수가 제공되지 않으면 현행구동기와 등록부가 현시된다. DOS 의 CD 지령과 류사하다.

문법 mcd [msdos **등록부**]

주요기발과 선택항목 없음

mcopy

설명 플로피디스크와 같은 UNIX 와 MS-DOS 파일체계들사이에 량쪽 방향으로 파일들을 복사한다. 이 지령의 마지막인수가 등록부일 때 한개의 등록부안에 여러개의 파일들을 복사할 수 있다. a: 와 같은 DOS 구동기지적자를 사용하여 DOS 파일을 복사하며 그렇지 않으면 Unix 파일체계로 인정한다. 인수로 한개의 DOS 파일만을 가지면 이 파일은 현재 Unix 등록부에 복사된다. DOS 의 COPY 지령과 류사하다.

문법 mcopy [-tnm] **원천파일 목적파일**
mcopy [-tnm] **원천파일** [**원천파일**…] **목적등록부**
mcopy [-tnm] MSDOS **원천파일**

주요기발과 선택항목

- **-m**: 복사될 때마다 파일이 수정되는것을 막는다.
- **-n**: Unix 파일들에 덧쓰기할 때 확증을 묻지 않는다.
- **-t**: 복사할 때 Unix 와 DOS 본문파일사이에서 본문파일들을 변환한다.

mdel

설명 MS-DOS 의 플로피디스크의 파일들을 지운다. DOS 의 DEL 지령과 류사하다.

문법 **mdel msdos 파일 [msdos 파일 …]**

주요기발과 선택항목 없음

mdeltree

설명 하나이상의 MS-DOS 등록부들을 지운다. DOS 의 DELTREE 지령과 류사하다.

문법 **mdeltree msdos 등록부 [msdos 등록부 …]**

주요기발과 선택항목 없음

mdir

설명 플로피디스크와 같은 MS-DOS 파일체계우에서 등록부의 내용을 현시한다. 파일이 지정되면 그 파일만을 현시한다. DOS 의 DIR 지령과 류사하다.

문법 **mdir [-w] msdos 등록부**

mdir [-a] [-f] [-w] msdos 파일 [msdos 파일 …]

주요기발과 선택항목

- **-a**: 규칙적인 파일은 물론 숨겨 진 파일을 현시한다.
- **-f**: 현시마감에 빈 공간없이 총 파일들을 현시한다.
- **-w**: 파일들을 광폭으로 현시한다.

message

설명 사용자의 우편함이나 이름이 붙었다면 우편서류철에 있는 통보문의 총 수를 현시한다.

문법 **message [서류철]**

주요기발과 선택항목 없음

mformat

설명 MS-DOS 의 플로피디스크를 양식화한다. DOS 의 FORMAT 지령과 류사하다.

문법 **mformat 구동기:**

주요기발과 선택항목 없음

mkdir

설명 하나이상의 등록부를 만든다.

문법 **mkdir [-p] [-m 방식] [--parents][--mode=방식] 등록부 …**

주요기발과 선택항목

- **-m/--mode 방식:** chmod 지령과 같은 그런 기호적인 현시법을 써서 등록부를 만들기 위한 방식을 설정한다. 이것이 지정되지 않으면 기정방식이 등록부에 할당된다.
- **-p/--parents:** 지정된 등록부의 어미들이 존재한다는것을 담보하며 임의의 필요한 어미등록부들을 만든다.

mkdosfs

설명 지정된 등록부우에서 MS-DOS 파일체계를 양식화한다.

문법 **mkdosfs [-c] [-F fat 크기] [-n 이름] 구동기**

주요기발과 선택항목

- **-c:** 양식화하기전에 나쁜블로크를 검사한다.
- **-f fat 크기:** 만들려는 파일배치형태를 지정한다. 이것은 12비트 FAT에 대해서는 12, 16비트 FAT에 대해서는 16이다. 프로그람은 사용자에게 가장 좋은 선택항목들을 선택하는데 보통 16이다.
- **-n 이름:** 지정된 이름에 대한 볼륨표식을 설정한다. 이름은 11문자이상 길게 할수 있다. 이름이 없으면 볼륨표식은 설정되지 않는다.

mke2fs

설명 두번째로 확장된 Linux의 파일체계를 양식화한다.

문법 **mke2fs [-c] [-m 비률] [-L 표식] 장치**

주요기발과 선택항목

- **-c:** 양식화하기전에 나쁜 블로크를 검사한다.
- **-L 표식:** 볼륨표식을 지정한것으로 설정한다.
- **-m 퍼센트:** 상급사용자들을 위하여 남겨 둔 블로크의 비률을 지정한다. 이것은 기정으로는 5%로 설정된다.

mkfs

설명 파일체계를 만든다(DOS에서 구동기의 양식화와 류사하

다). 파일체계를 위한 블로크들을 마음대로 지정할수 있다.

문법 **mkfs [-t 파일체계형] [-c] [-l 파일] 장치 [블로크들]**

주요기발과 선택항목

- -c: 양식화하기전에 나쁜 블로크에 대한 장치를 검사한다.
- -l **파일**: 지정된 파일로부터 장치에 대한 나쁜 블로크목록을 읽는다.
- -t **파일체계형**: 만들어야 하는 파일체계의 형을 지정한다. 기정파일체계형은 /etc/fstab 에서 장치를 위한 다른 방식을 찾을수 없다면 minix 이다.

mkpasswd

설명 한개의 통과암호를 만들고 그것을 사용자에게 마음대로 할당한다.

문법 **mkpasswd [-2] [-l 수] [-d 수] [-c 수] [-C 수][-p 파일][사용자]**

주요기발과 선택항목

- -2: 표준 U.S.건반으로 건입력할 때 오른손과 왼손의 문자들을 엇바꾸어 넣게 한다.
- -c **수**: 통과암호의 소문자최소수를 지정한다.
- -C **수**: 통과암호의 대문자최소수를 지정한다.
- -d **수**: 통과암호의 수자최소수를 지정한다.
- -l **수**: 통과암호의 문자수를 지정한다.
- -p **파일**: 통과암호설정에 사용되는 프로그람을 지정한다. 기정으로 이것은 /etc/yppasswd 이거나 그것이 없으면 /bin/passwd 이다.

주의 -2 기발을 사용하면 친숙한 관찰자들이 사용자가 무엇을 입력하는가를 보기 힘들다. 그러나 통과암호추축프로그람이 통과암호를 쉽게 추축할수 있게 한다.

mkswap

설명 교환구역으로서 장치를 설정한다. 파일체계의 크기는 필요하다면 블로크로 지정할수 있다.

문법 **mkswap [-c] 장치 [블로크크기]**

주요기발과 선택항목 -c: 교환파일체계를 만들기에 앞서 나쁜 블로크에 대한 장치를 검사한다.

주의 교환구획대신에 교환파일을 만드는것이 가능하다. 더 상세한 정보는 **mkswap** man 페지를 참고하여야 한다.

mlabel

설명 플로프디스크와 같은 MS-DOS 파일체계를 현시한다. 표식이 제공되지 않으면 사용자표식을 기다린다.

문법 mlabel [-cs] **구동기: 표식**

주요기발과 선택항목

- -c: 아무런 질문없이 사용자에 대한 현재 표식을 지운다.
- -s: 현재표식을 현시한다.

mmd

설명 플로프디스크와 같은 MS-DOS파일체계에 하나이상의 등록부를 만든다.

문법 mmd msdos **등록부** [msdos **등록부** …]

주요기발과 선택항목 없음

mmove

설명 MS-DOS파일이나 등록부를 옮긴다(또는 이름을 바꾼다.). 마지막인수가 등록부이면 모든 원천파일을 목적등록부에 옮긴다.

문법

mmove **원천파일 목적파일**
mmove **원천파일** [**원천파일** …] **목적등록부**

주요기발과 선택항목 없음

modprobe

설명 패턴이나 지정된 모듈대상파일에 기초하여 하나이상의 적재가능한 모듈들을 지정한다.

문법

modprobe **모듈** .o [symbol=**값** …]
modprobe -t **꼬리표 패턴**
modprobe -a -t **꼬리표 패턴**
modprobe -l [-t **꼬리표**] **패턴**
modprobe -r **모듈**
modprobe -c

주요기발과 선택항목

- -a: 성과적으로 적재하는 첫번째 모듈이 아니라 모든 모듈을 적재한다.

- -c: 구성정보를 현시한다.
- -l: 지정된 형태의 모든 모듈을 현시한다.
- -r **모듈**: 지정된 모듈탄창을 적재하지 않는다.
- -t **꼬리표**: 지정된 꼬리표로 현시된 모듈만을 적재한다.

more

설명

하나이상의 파일들을 한 화면씩 현시하며 파일에서 임의의 위치에로 탐색하여 뛰여 넘게 한다.

문법

more [-dlfs] [-**수**] [+**수**] [**파일** …]

주요기발과 선택항목

- -**수**: 화면당 행수를 설정한다.
- +**수**: 현시할 때 시작행을 지정한다.
- -d: 매 화면의 끝에서 사용자를 재촉한다.
- -f: 긴 행들이 겹치지 않고 한 행에 포함되게 한다.
- -l: Ctrl+L 이 양식바꾸기로 처리지 않게 한다.
- -s: 여러개의 공백행을 하나의 공백행으로 압축한다.

주의

파일을 볼 때 리용되는 지령들에 대한 상세한 정보는 more man 를 참고하여야 한다.

mount

설명

지정된등록부에 파일체계를 설치한다.

문법

mount -a [-rw] [-t **가상파일체계형**]

mount [-rw] [-o **선택항목**[,…]] **장치|등록부**

mount [-rw] [-t **가상파일체계형**] [-o **선택항들**] **장치 등록부**

주요기발과 선택항목

- -a: /etc/fstab 에 모든 파일체계들을 설치한다. 형이 -t 로 지정되면 /etc/fstab 에서 지정된 형의 파일체계만 적재한다.
- -o: 아래에 서술된 주의를 보면 된다.
- -r: 읽기전용방식으로 파일체계를 설치한다. 이것은 -o ro 와 같다.
- -t 파일체계형: 파일체계형을 지정한다.
- -w: 읽기쓰기방식으로 파일체계를 설치한다. 이것은 -o

rw 와 같다.

주의 -t 기발을 위한 가능한 파일체계형은 다음과 같다.

Minix	vfat	ufs
ext	proc	romfs
ext2	nfs	sysv
xiafs	iso9660	xenix
hpfs	smbfs	coherent
msdos	ncpfs	
umsdos	affs	

-o 기발을 위한 선택항목에 대해서는 리용가능한 여러가지 선택항목에 대한 상세한 설명을 주는 **mount** man 페지를 참고한다.

mrd

설명 하나이상의 MS-DOS 등록부를 삭제한다.

문법 **mrd msdos 등록부 [msdos 등록부 …]**

주요기발과 선택항목 없음

mren

설명 MS-DOS 의 파일이름을 갱신한다. 등록부의 이름도 갱신할 수 있다는것을 제외하고는 DOS 의 **REN** 지령과 류사하다.

문법 **mren 낡은이름 새이름**

주요기발과 선택항목 없음

mtype

설명 하나이상의 MS-DOS 파일내용을 현시한다. DOS 의 **TYPE** 지령과 류사하다.

문법 **mtype [-ts] msdos 파일 [msdos 파일 …]**

주요기발과 선택항목

- -s: 자료에서 높은 자리에 있는 비트를 삭제한다.
- -t: DOS 본문파일을 현시하기전에 UNIX 본문파일들로 변환한다.

mv

설명 파일의 이름을 갱신하고 옮긴다. 등록부가 마지막인수이면 지정된 모든 파일들은 그 등록부에로 옮겨 간다.

문법 **mv [-bfiu] [--backup] [--force] [--interactive]**

[--update] {**원천 목적|원천 … 등록부**}

주요기발과 선택항목

- -b/--backup: 옮기려는 파일의 여벌복사물을 만든다.
- -f/--force: 파일을 옮길 때 덧쓰기에 대해서는 물어보지 않고 이미 존재하는 파일들을 삭제한다.
- -i/--interactive: 임의의 존재하는 파일을 덧쓰기하기전에 재촉한다.
- -u/--update: 존재하는 파일의 수정시간이 같거나 더 새롭다면 덧쓰기를 하지 않는다.

netstat

설명

련결, 경로조종표 그리고 대면부통계를 포함하는 망상태정보를 현시한다. 선택항목이 제공되지 않으면 능동적인 소케트들의 목록이 현시된다.

문법

netstat [-Mnrs] [-c] [-i interface] [--interface] [--masquerade] [--route] [--statistics]

주요기발과 선택항목

- -c: Ctrl+C 가 눌리울 때까지 매 초간격으로 선택정보를 현시한다.
- -i [**대면부**] /--interface [**대면부**]: 지정된 대면부에 대한 정보를 현시하거나 혹은 아무것도 지정되지 않으면 모든 대면부에 대한 정보를 현시한다.
- -M/--masquerade: 가장된 대화조종목록을 현시한다.
- -n: 주콤퓨터, 포구, 사용자이름대신에 수값주소를 보여 준다.
- -r/--route: 핵심부경로조종표들을 현시한다.
- -s/--statistics: 망통계를 현시한다.

newgrp

설명

사용자가 새로운 그룹에 가입하면 사용자의 그룹 ID 를 변화시킨다. 그룹이 지정되지 않으면 그룹 ID 는 사용자의 가입그룹 ID 로 바꾼다.

문법

newgrp [**그룹**]

주요기발과 선택항목 없음

newusers

설명

새로운 사용자들의 목록을 포함하는 파일을 읽으며 사용자들을 만든다. 파일이 지정되지 않으면 사용자정보는 표준입

구로부터 제공된다.

문법 newusers [**파일**]

주요기발과 선택항목 없음

주의 newusers지령으로 읽은 파일류형은 다음의 요구를 포함하는 /etc/passwd와 같다.

- 통과암호를 암호화하지 않는다. 그것들은 사용자들의 회계가 만들어 질 때 암호화된다.
- 지정된 그룹이 없으면 새로운 그룹이 생성된다.
- 사용자홈등록부가 없으면 새로운 등록부가 만들어 진다. 그 이름을 가진 등록부가 이미 존재한다면 소유권은 새로운 사용자의것으로 바뀐다.
- 이 파일이 복호화된 통과암호를 포함하기때문에 그것을 항상 안전하게 보관하는것이 필요하다.

nisdomainname

설명 현재 NIS령역이름을 현시한다.

문법 nisdomainname

주요기발과 선택항목 없음

nslookup

설명 DNS 이름봉사기에 조회한다. 호상작용방식으로 실행할수 있다. 주콤퓨터이름이 제공되지 않으면 프로그람은 호상작용방식으로 들어 간다. 다른것이 지정되지 않는 한 기정으로 /etc/resolv.conf 에 지정된 DNS 봉사기를 사용한다. 봉사기를 지정하지만 지정된 주콤퓨터를 보지 않으려면 주콤퓨터대신에 봉사기를 제공해야 한다.

문법 nslookup [**주콤퓨터**|-[**봉사기**]]

주요기발과 선택항목 없음

주의 호상작용방식에서 쓸수 있는 지령들에 대한 소개는 nslookup man 페지를 보아야 한다.

passwd

설명 사용자의 통과암호를 변경시킨다. 뿌리사용자에 의해 실행될 때에 그것은 사용자이름을 인수로 제공함으로써 지정한 사용자의 통과암호를 바꾸는데 리용될수 있다.

문법 passwd [**사용자이름**]

주요기발과 선택항목 없음

paste

설명

하나이상의 파일들로부터 대응하는 행들을 붙인다. 한 행에 대응하는 행들을 순차적으로 현시하는데 이 행들은 꼬리표에 의해서 분리된다. 파일이름들이 주어 지지 않으면 입력은 표준입력으로 취해 진다.

문법

paste [-s] [-d **경계부호목록**] [--serial] [--delimiters **목록**] [**파일** …]

주요기발과 선택항목

- -d/--delimiters **목록**: 기정꼬리표문자대신 사용하려는 경계부호를 지정한다. 한개이상의 문자가 목록안에 있으면 문자들은 련속적으로 리용되는데 마지막문자가 리용된후 목록안에 있는 첫 문자로 돌아 간다.
- -s/--serial: 매 파일의 행이 아니라 다음파일의 행들에 따르는 파일의 행들을 붙인다.

pathchk

설명

파일이름들이 타당하며 편리한가를 검사한다. 특히 파일의 경로에 있는 모든 등록부들이 정당한 실행권한을 가지고 있는가 그리고 경로에서의 매 요소의 파일이름길이가 파일이름요소에 대한 최대길이보다 더 길지 않은가를 검사한다.

문법

pathchk [-p] [--potability] **파일** …

주요기발과 선택항목

-p/--portability: 실지 파일체계의 길이를 제한하지 않고 매 파일이름의 길이가 POSIX.1 표준을 위반하지 않는가를 검사한다. 파일이름에서 사용되는 문자들이 맞춤한가에 대해서도 검사한다.

pidof

설명

명명된 하나이상의 프로그람들의 프로쎄스 ID 들을 찾고 PID 들을 현시한다.

문법

pidof [-s] [-x] [-o **프로쎄스식별자**] [-o **프로쎄스식별자** …] **프로그람** [**프로그람** …]

주요기발과 선택항목

- -o **프로쎄스식별자**: 복귀된 목록으로부터 지정된 프로쎄스 ID 를 생략한다.
- -s: 하나의 PID 만을 준다.
- -x: 명명된 프로그람들은 물론 명명된 스크립트를 실행하는 쉘들의 PID 를 준다

ping

설명 망에서 호출가능한가를 보기 위하여 망가입자에게 응답요구파케트들을 보낸다.

문법 **ping [-R] [-c 파케트수][-d] [-i 초] 망가입자**

주요기발과 선택항목

- **-c 파케트수:** 지정된 파케트수를 전송한후 파케트발송을 중지한다.
- -d: 파케트들이 돌아 온것만큼 빨리 혹은 초당 100 번의 속도로 그것들을 전송한다. 더 큰수가 발생된다. 이 선택항목은 그것이 극히 큰 비중의 망통화밀도를 일으킬수 있기때문에 뿌리사용자에 의해서만 리용될수 있다. 이 선택항목을 사용할 때 주의해야 한다.
- **-i 초:** 매 파케트를 보내는 사이에 기다리는 시간을 초로 지정한다. 기정값은 1 초이다. 이 선택항목은 -f 선택항목과 함께 리용될수 없다.
- -R: 파케트를 전송하는 경로를 기록하고 복귀된 파케트들의 경로완충기를 현시한다.

pppstate

설명 ppp 동작의 통계를 현시한다.

문법 **pppstats [-a] [-v] [-r] [-z] [-c 계수값] [-w secs] [대면부]**

주요기발과 선택항목 없음

ps

설명 현재 실행중에 있는 프로쎄스에 대한 상태보고를 현시한다. 인수로 특수한 프로쎄스 ID 를 주면 ps 는 특수한 프로쎄스에 대한 정보를 현시한다. 선택항목이나 인수들이 없으면 ps 는 현재 사용자의 프로쎄스를 현시한다.

문법 **ps [lumaxwrf] [txx] [프로쎄스식별자 …]**

주요기발과 선택항목

- a: 현재 사용자의 프로쎄스들외에 다른 사용자들이 인정하는 프로쎄스들을 보여 준다.
- f: 다른 프로쎄스들을 자식으로 가진것을 보여 주는 나무모양의 프로쎄스들을 현시한다.
- l: 긴 형식의 정보를 현시한다.
- m: 보고서에 기억기정보를 현시한다.

- r: 실행되는 프로쎄스들만 현시한다.
- txx: xx 로 지정된 tty 에 의해 조종되는 프로쎄스들만을 현시한다.
- u: 사용자이름과 양식화의 시작시간을 비롯한 사용자형식으로 정보를 현시한다.
- w: 넓은 출력방식으로 정보를 현시하는데 한행에 그것들을 꼭 들어 맞추도록 하기 위하여 지령들을 자르는것을 금지시킨다. 선택항목으로 포함된 매 w 에 대하여 추가행은 지령을 현시하기 위하여 제공된다. 100 개이상의 w 선택항목들을 리용할수 있다.
- x: 말단들을 조종하지 않고 프로쎄스들을 보여 준다(이것은 시동프로쎄스가 실행되기까지의 기간에 쓰이는 데몬들을 보기 위하여 리용된다.).

pwck

설명	오유와 문제점에 대한 통파암호파일을 검사한다. 모든 입구점들의 형식은 매 마당에 유효한 정보가 나타나는가를 확인하기 위하여 검사된다. 그외에 중복된 항목들을 찾고 사용자에게 불충분하게 양식화되였거나 중복된 항목들을 지울수 있는 기회를 준다. 통파암호파일이나 그림자파일이 지정되지 않으면 기정값은 /etc/passwd 와 /etc/shadow 를 리용한다.
문법	pwck [-r] [**통파암호파일 그림자파일**]
주요기발과 선택항목	-r: 검사를 하지만 통파암호파일에 변화를 주지 않기 위하여 읽기전용방식으로 실행한다.

pwconv

설명	통파암호파일을 이미 존재하고 있는 그림자파일과 병합해서 그림자파일에 그것들의 항목들을 복사한다. 새로운 통파암호파일은 npassw 를, 새로운 그림자파일은 nshadow 를 호출한다.
문법	pwconv
주요기발과 선택항목	없음

pwd

설명	현재등록부의 이름을 현시한다.
문법	pwd
주요기발과 선택항목	없음

pwunconv

설명	그림자통파암호파일로부터 통파암호를 복귀한다. 새로운 통

파암호파일은 npasswd를 호출한다.

문법 pwunconv

주요기발과 선택항목 없음

quota

설명 사용자디스크의 사용분당 정보를 현시한다. 뿌리사용자는 지정한 사용자들과 그룹들을 지정할수 있고 그것들에 대한 보고서를 얻을수 있다. 뿌리사용자가 아닌 사용자들은 자기들이 인정하는 회계와 자기들이 속해 있는 그룹에 대한 정보를 보기만 할수 있다.

문법 quota [-guv] [**사용자|그룹**]

주요기발과 선택항목

- -g: 사용자가 속해 있는 그룹에 대한 그룹할당을 현시한다.
- -u: 이것은 사용자에 대한 할당정보를 현시하는 기정기발이다.
- -v:기억기가 할당되지 않은 파일체계의 할당을 현시한다.

quotacheck

설명 사용자나 그룹이 사용하는 디스크에 대한 파일체계를 훑어보고 결과를 quota.user 와 quota.group **의** 두개 파일에 출력한다.

문법 quotacheck [-g] [-u] [-a|**파일체계**]

주요기발과 선택항목

- -a: /etc/fstab 파일에 있는 모든 파일체계를 검사한다.
- -g: 개별적인 그룹 ID가 사용한 파일과 등록부에 대한 검사를 진행한다.
- -u: 개별적인 사용자 ID가 사용한 파일과 등록부에 대한 검사를 진행한다.

quotaoff

설명 하나이상의 파일체계에 대한 디스크사용할당을 할수 없게 한다.

문법 quotaoff [-g] [-u] [-a|**파일체계** …]

주요기발과 선택항목

- -a: /etc/fstab의 모든 파일체계에 대하여 할당할수 없게 한다.
- -g: 지정된 파일체계들에 대하여 그룹할당을 할수

없게 한다.

- -u: 지정된 파일체계들에 대하여 사용자할당을 할수 없게 한다.

quotaon

설명
하나이상의 파일체계에 대한 디스크사용할당을 할수 있게 한다.

문법
quotaon [-g] [-u] [-a|**파일체계** …]

주요기발과 선택항목

- -a: /etc/fstab의 모든 파일체계를 할당할수 있다.
- -g: 지정된 파일체계들에 대하여 그룹할당을 할수 있게 한다.
- -u: 지정된 파일체계들에 대하여 사용자할당을 할수 있게 한다.

rcp

설명
하나이상의 파일을 두 체계사이에서 원격복사한다. 마지막 인수가 등록부이면 모든 다른 파일인수들이 그 등록부에 복사된다.

문법
rcp [-px] **파일** …

주요기발과 선택항목

- -p: 가능한껏 원천파일의 수정시간들과 방식들을 보관한다.
- -x: 모든 복사물에 대하여 DES암호계로 암호화한다.

주의
원격파일과 등록부들은 remoteuser@remote-host:/path/to/file 형식으로 지정된다.

rdate

설명
망에서 하나이상의 주콤퓨터로부터 현행시간을 회복시키고 회복한 시간을 현시한다.

문법
rdate [-p] [-s] **주콤퓨터** …

주요기발과 선택항목

- -p: 원격체계로부터 회복한 시간을 현시한다(이것은 기정동작이다.).
- -s: 망으로부터 회복된 시간에 기초하여 국부체계의 시간을 설정한다.

rdist

설명 가능한껏 소유자, 방식, 갱신시간을 보존하면서 여러 주콤퓨터에서 꼭같은 복사물을 관리하기 위하여 파일들을 원격으로 배포한다. 원격주콤퓨터에서 배포등록부가 지정되지 않으면 원천파일들은 원격체계우에서와 같은 위치에 놓인다.

문법 rdist -c **파일|등록부** … [**등록가입@**] **주콤퓨터** [**:등록부**]

주요기발과 선택항목 -c: 원격체계에 배포되는 파일의 목록을 지정한다.

주의 Rdist는 구성파일로부터 분산을 조종하게 한다. 이것은 많은 대부분의 기발들과 선택항목들이 리용될수 있게 한다. 이 도구의 완전한 서술에 대해서는 rdist man를 참고한다.

repquota

설명 하나이상의 파일체계에 대한 디스크사용할당개요를 현시한다.

문법 repquota [-gu] [-a|**파일체계** …]

주요기발과 선택항목

- -a: /etc/fstab**의** 모든 파일체계에 대한 보고서를 현시한다.
- -g: 지정된 파일체계에 대한 그룹사용할당보고서를 현시한다.
- -u: 지정된 파일체계에 대한 사용자사용할당보고서를 현시한다.

rlogin

설명 원격주콤퓨터에 등록가입한다.

문법 rlogin [-Kx] [-l **사용자이름**] **주콤퓨터**

주요기발과 선택항목

- -K: 모든 Kerberos 인증을 무시한다.
- -l **사용자이름**: 사용자가 rlogin을 실행하지 않고 지정된 사용자이름하에서 접속이 가입되여야 한다는것을 가리킨다.
- -x: 원격체계로 가입될 때 전송되는 모든 자료를 DES로 암호화한다.

rm

설명 하나이상의 파일이나 등록부를 지운다.

문법 rm [-firR] [--force] [--interactive] [--recursive] **파일**

|**등록부** …

주요기발과 선택항목

- -f/--force: 파일을 지우는 사용허가에 대하여 사용자에게 묻지 않는다. 이것은 뿌리사용자로 리용되면 위험하다.
- -i/--interactive: 매 파일을 지우기전에 항상 사용자에게 묻는다.
- -r/-R/--recursive: 등록부들의 내용을 반복하여 삭제한다.

rmdir

설명

등록부입구점들을 지운다.

문법

rmdir [-p] [--parents] **등록부** …

주요기발과 선택항목

-p/--parents: 어미등록부가 빈것으로 되게 하는 등록부들을 더는 지우지 않고 지령행에 명백하게 지정된 어미등록부는 물론 그 등록부들을 삭제한다.

rmmod

설명

하나이상의 적재모듈들을 해제한다.

문법

rmmod [-as] **모듈** …

주요기발과 선택항목

- -a: 사용되지 않는 모든 모듈들을 삭제한다.
- -s: 모든 출력을 현시장치가 아니라 체계로그에 보낸다.

route

설명

IP 경로표를 현시하거나 바꾼다. 선택항목들이 지정되지 않으면 경로표를 현시한다.

문법

route add [-net|-host] **목표** [netmask **망마스크**] [gw **관문**] [[dev] If]

route del [-net|-host] **목표** [gw **관문**] [netmask **망마스크**] [[dev] If]

주요기발과 선택항목

- add: 경로추가를 나타낸다.
- del: 경로삭제를 나타낸다.
- [dev] If: 경로가 지정된 대면부에로 접속되게 한다.

- gw GW: 경로를 위한 관문을 지정한다.
- -host: 목적지가 주콤퓨터임을 나타낸다.
- -net: 목적지가 망임을 나타낸다.
- netmask Nm: 경로에 대한 망마스크를 지정한다.

rsh

설명　원격체계우에 쉘을 연다. 지령이 제공되면 그 지령은 원격주콤퓨터우에서 실행되며 결과를 반환받고 접속을 끝낸다.

문법　rsh [-Kx] [-l **사용자이름**] **주콤퓨터이름** [**지령**]

주요기발과 선택항목

- -K: Kerberos 인증을 무시한다.
- -l **사용자이름**: rsh 를 실행하는 사용자보다 다른 사용자로서 원격주콤퓨터에 접속하려고 한다.
- -x: 두 주콤퓨터사이에 보내지는 모든 전송자료에 대한 DES 암호화를 할수 있게 한다.

runlevel

설명　체계의 현재와 이전 실행준위를 현시한다.

문법　runlevel

주요기발과 선택항목　없음

rup

설명　하나이상의 원격체계들의 상태를 현시한다.

문법　rup [-dhlt] [**주콤퓨터** …]

주요기발과 선택항목

- -d: 매개 주콤퓨터에 대한 지방시간을 현시한다.
- -h: 항목들을 주콤퓨터이름으로 분류한다.
- -l: 항목들을 부하률로 분류한다.
- -t: 항목들을 콤퓨터가 작용하는 기간으로 분류한다.

rusers

설명　국부망에 있는 한개이상의 콤퓨터들에 누가 등록가입되였는가를 현시한다. 주콤퓨터가 지정되지 않으면 국부망우의

모든 콤퓨터들에 등록가입된 모든 사용자들을 현시한다.

문법	rusers [-l] [**주콤퓨터이름**]
주요기발과 선택항목	-l: 사용자이름, 주콤퓨터이름, 그 사용자가 리용한 tty 그리고 등록가입시간, 기타 정보를 포함하는 긴 형식으로 결과를 현시한다.

rwho

설명	국부망의 모든 콤퓨터들에 등록가입된 사용자들의 목록을 현시한다.
문법	rwho
주요기발과 선택항목	없음

rwhod

설명	rwho 의뢰자로부터 들어 오는 요구들에 대답한다.
문법	rwhod
주요기발과 선택항목	없음

setclock

설명	콤퓨터의 하드웨어시간을 현재체계시간으로 설정한다.
문법	setclock
주요기발과 선택항목	없음

setkeycodes

설명	주사코드-건반코드대응표에 대응하는 건배치표를 적재한다. 인수들은 첫번째가 건의 주사코드이고 두번째가 그에 대응하는 건반코드인 쌍으로 제공된다.
문법	setkeycodes **주사코드 건반코드** …
주요기발과 선택항목	없음
주의	주사코드와 건반코드를 지정하는 방법을 알려면 setkeycodes man 페지를 참고하여야 한다.

showkey

설명	건에 의하여 발생하는 주사코드들과 건반코드들을 현시한다. 프로그람은 마지막건이 눌리운 다음에도 10 초동안 동작한다.

문법　　showkey [-sk --scancodes --keycodes]

주요기발과 선택항목

- -k/--keycodes: 건반코드들을 현시한다.
- -s/--scancodes: 주사코드들을 현시한다.

showmount

설명　　NFS 봉사기에서 설치의 현재상태를 현시한다. 주콤퓨터가 지정되지 않으면 특수한 주콤퓨터에서의 설치만을 현시한다.

문법　　showmount [-ade] [--all] [--directories] [--exports] [**주콤퓨터**]

주요기발과 선택항목

- -a/--all: host:directory 형식으로 리용하는 의뢰자의 주콤퓨터이름과 설치된 등록부를 다 현시한다.
- -d/--directories: 등록부들만을 현시한다.
- -e/--exports: 반출된 등록부들에 대한 봉사기목록을 현시한다.

shutdown

설명　　체계의 전원을 끄기전에 등록된 가입에서 탈퇴하기 위하여 일정한 지연을 주고 경고통보문을 내보내면서 체계를 안전하게 닫는다. 체계닫기는 지정된 시간내에 일어 나며 그 시간이 되면 즉시 전원이 차단된다.

문법　　shutdown [-rkhc] **시간** [**경고**]

주요기발과 선택항목

- -c: 실행의 체계닫기를 취소한다.
- -h: 체계닫기후에 끝낸다.
- -k: 경고통보문을 보내지만 체계를 닫지는 않는다.
- -r: 체계닫기후에 다시 기동한다.

sort

설명　　하나이상의 본문파일에 있는 행들을 분류하고 결과를 현시한다. 파일이 지정되지 않으면 표준입구로부터 취해진 자료가 분류된다. 분류된 자료는 표준출구에 현시된다.

문법　　sort [-cu] [-t separator] [-o **파일**] [-T tempdir] [-bdfMnr] [+**위치** 1 [-**위치** 2]] [-k **위치** 1[,**위치** 2]] [**파일** …]

주요기발과 선택항목

- -b: 분류건반을 찾을 때 행안에 있는 앞선 공백을 무시한다.
- -c: 입구자료가 분류되였는가를 검사하고 분류되지 않았으면 오유통보문을 내보낸다. 실지 분류는 진행되지 않는다.
- -d: 분류처리에서 문자, 수자, 공백을 제외한 모든 기호들을 무시한다.
- -f: 분류처리중에 나타나는 소문자는 대문자로 바꾼다.
- -k **위치 1**[,**위치 2**]: 분류건반으로 리용하려는 마당을 지정한다. 마당은 **위치 1**에서 시작하여 **위치 2** 혹은 그 행의 마지막까지 실행한다. 마당과 문자위치는 령으로부터 시작하여 지정된다.
- -M: 월을 분류한다. 즉 3개의 문자에 따르는 0개이상의 공백으로 시작되는 임의의 문자렬들을 대문자로 바꾸고 그것들의 월이름이 함축된것처럼 분류한다.
- -n: 문자렬이 선택적인 기호에 따르는 0개이상의 공백들로 시작되는 문자렬들을 수자적으로 비교한다.
- -o **파일**: 표준출구대신에 지정된 파일에 결과를 출구한다.
- +**위치 1** [-**위치 2**]: 분류키로 리용하는 마당을 지정한다. 이 마당은 **위치 1**에서 시작하여 **위치 2** 혹은 그 행의 마지막까지 실행한다.
- -r: 분류순서를 반대로 한다.
- -t **분리기호**: 지정된 분리기호가 매개 행에 있는 분류건반들을 찾기 위한 마당분리기호로 쓰인다는것을 지정한다.
- -u: 두 행들이 꼭 같을 때 첫번째 행만을 출구한다.

statserial

설명

포구의 핀들에 대한 신호를 현시하여 직렬포구의 상태와 handshaking 행의 상태를 보여 준다. 장치가 지정되지 않으면 기정으로 MODEM 환경변수의 값으로 되거나 혹은 변수가 설정되지 않으면 /dwv/cual 의 값으로 된다. 프로그람은 계속 순환하면서 Ctrl+c 를 누를 때까지 초당 갱신상태를 제공한다.

문법

statserial [-n|-d|-x] [**장치**]

주요기발과 선택항목

- -d: 10진수로 포구의 상태를 현시한다.

- -n: 순환할수 없게 하며 한개 상태만을 현시한다.
- -x: 16 진수로 포구의 상태를 현시한다.

su

설명

다른 사용자와 그룹 ID 하에서 새로운 쉘을 실행한다. 사용자가 지정되지 않으면 새로운 쉘은 뿌리사용자와 같이 실행된다.

문법

su [-flmp] [-c **지령**] [-s **쉘**] [--login] [-fast] [--preserve-environment] [--command=**지령**] [--shell=**쉘**] [-] [**사용자**]

주요기발과 선택항목

- -c command/--command=**지령**: 호상작용방식으로 쉘을 실행하지 않고 단일지령행으로 지정된 지령을 쉘에 넘긴다.
- -f/--fast: c 쉘과 확장된 c 쉘의 경우에 파일이름패턴확장을 무시하는 쉘에 -f 선택항목을 넘긴다.
- -/-l/--login: 새로운 쉘이 가입쉘로 되게 한다. 이것은 새로운 환경변수가 설정되고 경로가 바뀌며 현행등록부가 사용자의 홈등록부로 절환된다는것을 의미한다.
- -m/-p/--preserve-environment; HOME, USER, LOGNAME, SHELL 환경변수들이 변하지 못하게 한다.
- -s shell/--shell=**쉘**: 통과암호파일에 포함된 기정쉘대신에 지정된 쉘을 실행시킨다.

sync

설명

디스크고속완충기를 물리디스크에 보관한다. 이것은 임의의 변화된 정보를 디스크에 복사하게 한다.

문법

sync

주요기발과 선택항목

없음

tail

설명

하나이상의 파일에 대한 마지막부분을 현시한다. 기정으로 다른것이 지정되지 않는 한 매 파일들의 마지막 10 개 행들이 현시된다. 파일이름이 지정되지 않으면 표준입구로부터 자료를 읽고 파일에서와 같은 규칙에 따라 마지막토막의 자료를 현시한다.

문법

tail [-c **수** [bkm]] [-n **수**][-q] [--bytes **수**[bkm]] [--lines **수**] [--quiet] [--silent] [**파일** …]

주요기발과 선택항목

- -c/--bytes 수: 매 파일의 끝으로부터 지정된 바이트수를 현시한다. 수뒤에는 512byte 블로크에 대해서는 b, Kbyte에 대해서는 k, Mbyte에 대해서는 m가 붙을수 있다.
- -n/--lines 수: 매 파일의 끝에서부터 지정된 행수를 현시한다.
- -q/--quite/--silent: 여러 파일을 처리할 때 파일이름 머리부를 현시하지 못하게 한다.

tar

설명

하나이상의 파일이나 등록부에 대한 문서고파일들을 만든다.

문법

tar [-crtuxz] [-f **타르파일**] [--file **타르파일**] [--create] [--delete] [--preserve] [--append] [--same-owner] [--list] [--wpdate] [--extract] [--get] [--gzip] [**파일**| **등록부** …]

주요기발과 선택항목

- -c/--create: 새로운 문서고를 만든다.
- -delete: 이미 있는 문서고에서 파일들을 지운다.
- -f tar **파일**/--file tar **파일**: 문서고에서 만들어 지거나 읽어 진 파일의 이름을 지정한다.
- --preserve: 문서고에 있는 같은 파일들에 대한 허가들과 일정을 보존한다.
- -r/--append: 이미 있는 문서고에 파일들을 추가한다.
- --same-owner: 푼 파일들이 자기의 원래사용자를 보존한다는것을 나타낸다.
- -t/--list: 문서고내용목록을 현시한다.
- -u/--update: 문서고에 복사된것보다 더 새로운 파일들만을 이미 있는 보관고에 추가한다.
- -x/--ectract/--get: 이미 있는 문서고로부터 파일들을 푼다.
- -z/--gzip/--gunzip: 문서고에 넣을 때와 꺼낼 때 gzip를 써서 문서고를 려과한다.

timeconfig

설명

시간파라메터들을 구성한다. 시간대역이 주어 지면 체계시간대역을 지정된 시간대역으로 바꾼다. 한편 리용할수 있는 시간대역들의 목록을 현시한다.

문법 timeconfig [--utc] [**시간대역**]

주요기발과 선택항목 --utc: 체계시계가 만국표준시간/그리니치표준시간으로 동작한다고 가정한다.

timed

설명 시간을 국부망에 있는 다른 콤퓨터의 시간으로 동기시킬수 있는 시간봉사기데몬을 실행한다.

문법 timed [-M] [-i **망**] [-n **망**]

주요기발과 선택항목

- -i **망**: timed로 만든 임의의 기정선택을 무시하고 봉사기가 어느 망에 속하는가를 지정한다.
- -M: 주봉사기가 파괴되면 주시간봉사기의 일감을 수행할수 있게 준비한다.
- -n **망**: 지정된 망을 유효한 망의 목록에 첨부한다.

timedc

설명 timed데몬을 조종한다.

문법 timedc [clockdiff **주콤퓨터** …|msite[**주콤퓨터** …] lelection **주콤퓨터**]

주요기발과 선택항목

- clockdiff **주콤퓨터** …: 체계시계와 지정된 주콤퓨터의 시간사이 차이를 계산한다.
- election **주콤퓨터**: 선택박자계수기를 다시 설정하고 시간마스터가 슬래브가운데서 선택되였다는것을 담보한다.
- msite [**주콤퓨터** …]: 지정된 주콤퓨터나 주콤퓨터를 위한 주시간봉사기를 보여 준다. 주콤퓨터들이 지정되지 않으면 현재체계를 위한 주봉사기를 보여 준다.

top

설명 체계에서 실행하는 프로쎄스의 정기갱신보고를 현시한다.

문법 top [d **지연**] [q] [c] [S] [s]

주요기발과 선택항목

- c: 지령이름대신에 프로쎄스의 완전한 지령행을 현시한다.
- d **지연**: 갱신사이의 지연을 초로 지정한다.
- q: 아무런 지연이 없이 갱신되게 한다. 뿌리사용자

가 이 선택항목을 리용하면 top 는 가장 높은 주기로 실행한다.

- s: 위험한 호상작용지령들의 사용을 방지하는 보호방식에서 실행한다.
- S: 통계는 루적방식으로 현시된다. 즉 총체적으로 어떤 프로쎄스와 그의 닫긴 새끼프로쎄스에 대한 CPU 시간을 기록한다.

주의

top 가 실행될 때 사용될수 있는 지령목록과 보고에서 여러 마당들의 서술에 대해서는 top man 페지를 읽어야 한다.

touch

설명

파일의 내용을 변화시키지 않으면서 파일의 발신날자와 시간을 변화시킨다. 파일이 존재하지 않으면 크기가 령인 파일을 만든다. 기정값으로는 현재 시간을 새로운 파일의 발신날자와 시간으로 리용한다.

문법

touch [-acm] [-t MMDDhhmm[[CC] yy][.ss]] [--time=**호출시간**][--time=**호출**] [--time=**사용**] [--time=**수정시간**] [--time=**수정**] [--no-create] **파일** …

주요기발과 선택항목

- -a/--time=**호출시간**/--time=**호출**/--time=use: 호출시간만을 변경시킨다.
- -c/--no-create: 존재하지 않는 파일을 만들지 않는다.
- -m/--time=**수정시간**/--time=**수정**: 갱신시간만을 변화시킨다.
- -t MMDDhhmm [[CC]yy][.ss]: 발신날자와 시간을 지정된 월, 일, 시, 분, 지정된 세기, 년 혹은 초로 설정한다. 이것은 현행시간을 쓰는 기정값을 무시한다.

traceroute

설명

망에 원격주콤퓨터를 가입시키기 위한 파케트이행경로들을 현시한다.

문법

traceroute [-ir] **주콤퓨터**

주요기발과 선택항목

- -i: 출구되는 파케트에 대한 망대면부를 지정한다. 이것은 하나이상의 망대면부를 가진 체계에서 유용하다.
- -r: 표준경로선택표를 무시하고 가입된 주콤퓨터에게 직접 보내려고 한다.

true

설명

아무것도 하지 않고 성공적인 탈퇴상태를 복귀한다.

문법 true

주요기발과 선택항목 없음

umount

설명 설치된 파일체계를 해체한다. 파일체계는 자기의 장치구동프로그람이름, 등록부이름이든가 혹은 자기의 망경로로 지정한다.

문법 umount -r **장치구동프로그람 |등록부| 경로 …**

주요기발과 선택항목 -r: 해체가 실패하면 읽기전용방식으로 파일체계를 다시 설치할수 있다.

uname

설명 선택된 체계정보를 현시한다. 선택항목이 제공되지 않으면 조작체계의 이름이 현시된다. 정보에 대한 여러 부분이 요구될 때 현시순서는 항상 조작체계, 망주콤퓨터이름, 조작체계공개물, 조작체계판본 그리고 콤퓨터의 형이다.

문법 uname [-snrvma] [--sysname] [--nodename] [--release] [--machine] [--all]

주요기발과 선택항목

- -a/-all: 모든 정보를 현시한다.
- -m/--machine: 콤퓨터형을 현시한다(즉 하드웨어형).
- -n/--nodename: 콤퓨터의 망주콤퓨터이름을 현시한다.
- -r/--release: 조작체계공개물을 현시한다.
- -s/sysname: 조작체계의 이름을 현시한다. 이것은 선택항목이 지정되지 않았을 때 기정동작이다.
- -v: 조작체계의 판본을 현시한다.

uncompress

설명 compress프로그람으로 압축된 파일을 푼다. 파일이 지정되지 않으면 표준입력이 풀린다.

문법 uncompress [-c] [**파일** …]

주요기발과 선택항목 -c: 압축된 낡은 파일을 덧쓰기하지 않고 압축되지 않은 자료를 표준출구에 보낸다.

unzip

설명 ZIP 문서를 조종하고 푼다.

문법 unzip [-cflptuz] [-d exdir] **파일** [.zip]

주요기발과 선택항목

- -c: 풀어 지는 매 파일의 이름을 현시하면서 표준출구에로 넣는다.
- -d exdir: 현재 등록부대신에 지정된 등록부에 대한 문서들을 푼다.
- -f: 이미 존재하고 있는 파일의 판본보다 더 새것인 파일만을 푼다.
- -l: 풀지 않고 문서고의 내용을 현시한다.
- -p: 파일이름과 같은 자료를 보내지 않고 표준출구에로 파일을 풀어 낸다.
- -t: 문서고에 있는 파일들의 완전성을 검사한다.
- -u: 푼 등록부에 이미 존재하지 않는 파일은 물론 이미 존재하는 파일의 판본보다 더 새로운 파일을 푼다.
- -z: 문서고해설을 현시한다.

주의 이것은 많은 개정물들을 지원하는 강력한 프로그람이다. 상세한것은 unzip man 페지를 보아야 한다.

uptime

설명 체계가 실행되고 있는 시간의 길이를 현시한다.

문법 uptime

주요기발과 선택항목 없음

useradd

설명 사용자를 체계에 추가한다.새로운 사용자들에 대한 기정값을 엇갈아 변화시킬수 있다. 선택항목이 제공되지 않으면 프로그람은 새 사용자에 대한 현재 기정값들을 현시한다.

문법 useradd [-d **홈등록부**] [-e **만기자료**] [-f **정지시간**] [-g **초기그룹**] [-G **그룹**[,…]] [-s **쉘**] [-u **사용자식별자** [-o]] **사용자이름**

useradd -D [-g **기정그룹**] [-b **기정홈등록부**] [-f **기정정지시간**] [-e **기정만기자료**] [-s **쉘**]

주요기발과 선택항목

- -b **기정홈등록부**: 기정홈등록부머리부를 지정된 경로로 설정한다. -D가 쓰일 때만 리용된다.
- -d **홈등록부**: 기정홈등록부대신에 사용자에 대하여 지

정된 홈등록부를 사용한다.

- –D: 새로운 사용자를 만들기보다는 새로운 사용자들에 대한 기정값들을 변화시켜야 한다는것을 나타낸다.
- –e **만기자료**: 회계에 대한 만기자료를 지정한다. 자료는 MM/DD/YY 의 형태로 제공된다. –D 가 사용될 때 –e 는 모든 새로운 통과암호에 대한 기정만기자료를 자료형태로가 아니라 날자수로 설정하는데 리용된다.
- –f **정지시간**: 일단 통과암호가 만기되면 통과암호가 영구적으로 무시되기전에 지정된 총 시간이 경과되여야 한다는것을 나타낸다. –D 가 사용될 때 –f 는 만기된후 통과암호가 무시되기전에 기정총시간을 설정하는데 리용된다.
- –g **초기그룹**: 사용자들에 대한 기정등록가입그룹을 지정한다. –D 를 사용할 때 –g 는 모든 새로운 사용자들에 대한 기정그룹을 지정한다.
- –G **그룹**[,…]: 사용자가 속해야 할 다른 그룹을 지정한다.
- –s **쉘**: 사용자에 대한 기정쉘을 지정한다. 제공되지 않으면 새로운 사용자에 대한 기정쉘을 사용한다. –s 를 –D 와 함께 사용하면 새로운 사용자에 대한 기정쉘을 지정한다.
- u *uid* [–o]: ID 를 자동적으로 할당하는것보다 사용자가 규정하는것이 더 낫다. 그 결과는 –o 기발을 리용하지 않는 한 오직 유일한 값이여야 한다.

userdel

설명　사용자의 회계를 지운다.

문법　userdel [–r] **사용자**

주요기발과 선택항목　–r: 회계를 삭제할 때 사용자의 홈등록부를 삭제한다.

주의　만일 홈등록부삭제를 선택하면 회계를 삭제할 때 삭제되지 않는 홈등록부의 밖에서 사용자가 다른 파일들을 보존한다. 이것들은 수동적으로 삭제해야 한다. 실례로 이러한 파일에는 사용자의 우편함이 있어야 한다.

usermod

설명　이미 존재하는 사용자의 회계에 대한 설정을 수정한다.

문법　usermod [–d **홈등록부**] [–m] [–e **만기자료**] [–f **정지시간**] [–g **초기그룹**] [–G **그룹**[,…]] [–l **등록가입이름**] [–s **쉘**] [–u **사용자식별자** [–o]] **등록가입**

주요기발과 선택항목

- **-d 홈등록부 [-m]**: 사용자의 홈등록부를 지정한대로 바꾸고 -m 선택항목이 사용되면 현재 홈등록부를 새로운 위치에 옮긴다.
- **-e 만기자료**: 회계가 무시된후 그에 대한 새로운 만기자료를 설정한다. 자료는 MM/DD/YY 형태로 되여야 한다.
- **-f 정지시간**: 통과암호가 영구적으로 무시될 때 통과암호가 만기된후 날들에 대한 새로운 설정을 제공한다.
- **-g 초기그룹**: 사용자에 대한 새로운 가입집단을 정의한다.
- **-G 그룹[,…]**: 사용자가 다른 그룹의 성원이여야 한다는것을 나타낸다. 사용자가 현재 이 목록에 없는 그룹의 성원이라면 그것들은 그룹으로부터 삭제된다.
- **-l 등록가입이름**: 사용자의 가입이름을 바꾼다.
- **-s 쉘**: 사용자의 기정쉘을 지정한대로 바꾼다.
- **-u 사용자식별자**: 사용자 ID 를 지정한대로 바꾼다.

users

설명

체계우에 있는 현재 사용자들의 사용자이름을 현시한다. 이것은 보통 /etc/utmp 의 내용을 보면서 찾는다. 파일이 지정되면 프로그람은 그 파일안에서 정보를 본다.

문법

users [**파일**]

주요기발과 선택항목

없음

uudecode

설명

uudecode 로 만든 ASCII 파일을 초기 2 진파일로 다시 만들도록 복호화한다. 기정적으로 복호파일이름은 부호화된 파일의 초기이름이다. 복호화하려는 파일이 제공되지 않으면 표준입력을 복호화한다.

문법

uudecode [-o **출력파일**] [**파일** …]

주요기발과 선택항목

-o **출력파일**: 결과복호파일에 대하여 대리이름을 지정한다.

uuencode

설명

2 진파일을 리용할수 없는데(일련의 전자우편쏘프트웨어들과 같은)서도 리용될수 있는 형태로 2 진파일을 부호화한다.

파일이 제공되지 않으면 표준입력을 부호화한다.

문법 uuencode [**파일**]

주요기발과 선택항목 없음

vipw

설명 EDITOR 환경변수에서 지정된 편집프로그람을 리용하여 체계통과암호파일을 편집한다.

문법 vipw

주요기발과 선택항목 없음

vmstat

설명 가상기억기에 대한 총계를 보고한다.

문법 vmstat [**지연** [**계수**]]

주요기발과 선택항목

- **계수**: 보고를 반복하는 회수를 나타낸다. 지정되지 않으면 보고는 Ctrl+c 로 중단될 때까지 계속 반복한다.
- **지연**: 보고를 얼마나 자주 반복하는가를 초로 나타낸다. 지정되지 않으면 한번만의 보고를 제공한다.

주의 vmstat 에 의해서 발생되는 보고의 상세한것에 대해서는 vmstat man 페지를 읽어보아야 한다..

w

설명 현재 실행하고 있는 사용자의 과제목록을 현시한다. 사용자가 지정되면 그 사용자의 과제만 현시한다.

문법 w[**사용자**]

주요기발과 선택항목 없음

wc

설명 하나이상의 문서에 대한 바이트수, 단어수, 행수를 현시한다. 여러개의 파일이름이 제공되면 매 파일은 분리되여 계수되고 현시되며 그다음 축적된 총수를 현시한다. 파일이 지정되지 않으면 표준입력이 계수된다.

문법 wc [-clw] [--bytes] [--chars] [--lines] [--words] [**파일 …**]

주요기발과 선택항목

- -c/--bytes/chars: 바이트수만 현시한다.
- -l/--lines: 행수만 현시한다.
- -w/--words: 단어수만 현시한다.

주의 결과는 문자, 단어, 행의 순서로 현시된다. 값들은 공백으로 분리된다.

whereis

설명 하나이상의 지령에 대한 2진코드, 원천코드, man페지들을 배치한다.

문법 whereis [-bms] [BMS **등록부** … -f] **파일** …

주요기발과 선택항목

- -b: 2진파일만을 탐색한다.
- -B **등록부** …-f: 지정된 등록부에서만 2진파일을 탐색한다. -f 기발은 등록부목록의 끝과 파일인수목록의 시작을 서술하기 위해서 필요하다.
- -m: man페지만을 탐색한다.
- -M **등록부** -f: man페지를 위한 지정된 등록부만을 탐색한다. -f 기발은 등록부목록의 끝과 파일인수목록의 시작을 서술하기 위하여 필요하다.
- -s: 원천코드만을 탐색한다.
- -S **등록부** … -f: 원천코드를 위해 지정된 등록부만을 탐색한다. -f 기발은 등록부목록의 끝과 파일인수목록의 시작을 서술하기 위해서 필요하다.

주의 whereis 지령은 최소한 프로그람들을 배치하려는 다음의 등록부들을 탐색한다.

/bin

/usr/bin

/etc

/usr/etc

/sbin

/usr/sbin

/usr/games

/usr/games/bin
/usr/emacs/etc
/usr/lib/emacs/19.22/etc
/usr/lib/emacs/19.23/etc
/usr/lib/emacs/19.24/etc
/usr/lib/emacs/19.25/etc
/usr/lib/emacs/19.26/etc
/usr/lib/emacs/19.27/etc
/usr/lib/emacs/19.28/etc
/usr/lib/emacs/19.29/etc
/usr/lib/emacs/19.30/etc
/usr/Tex/bin
/usr/tex/bin
/usr/interviews/bin/LINUX
/usr/bin/X11
/usr/X11/bin
/usr/X11R5/bin
/usr/X11R6/bin
/usr/X386/bin
/usr/local/bin
/usr/local/etc
/usr/local/sbin
/usr/local/games
/usr/local/games/bin
/usr/local/emacs/etc
/usr/local/Tex/bin
/usr/local/tex/bin
/usr/local/bin/X11
/usr/contrib
/usr/hosts

/usr/include

/usr/ginclude

which

설명 하나이상의 프로그람의 완전한 경로이름을 현시한다. PATH 환경변수에 의하여 지정된 경로우의 프로그람만을 현시한다.

문법 which **프로그람** …

주요기발과 선택항목 없음

whoami

설명 현재 유효한 사용자 ID를 현시한다.

문법 whoami

주요기발과 선택항목 없음

ypdomainname

설명 체계의 NIS 령역이름을 현시한다.

문법 ypdomainname

주요기발과 선택항목 없음

zcat

설명 하나이상의 압축된 파일을 풀고 표준출력에 결과를 현시한다. 파일이 지정되지 않으면 표준입력을 풀고 현시한다.

문법 zcat [-f][**파일** …]

주요기발과 선택항목 -f/--force: 대응하는 파일이 이미 존재할 때에도 풀기를 하고 풀린 파일로 덧쓰기한다.

zgrep

설명 지정된 패턴에 대하여 하나이상의 압축된 파일을 탐색한다.

문법 zgrep [**선택항목**] **패턴파일** …

주요기발과 선택항목 없음

주의 패턴문법의 가능한 항목에 대한 목록과 해설에 대하여서는 grrep 지령을 참고해야 한다.

zip

설명 　하나이상의 파일과 등록부로부터 ZIP 문서고를 만든다.

문법 　**zip [-efFgmrsu@] [zip 파일 [파일 1 파일 2…]]**

주요기발과 선택항목

- **-@**: 표준입력으로부터 문서고에 넣게 되는 파일들의 목록을 접수한다.
- **-e**: 통과암호를 기다린후에 문서고를 암호화한다. 통과암호는 문서고로부터 파일을 뽑기 위하여 필요하다.
- **-f**: 파일이 현행문서고에 있는 파일보다 더 새것이라면 존재하는 문서만 항목들을 교체한다.
- **-F**: 손해를 입은 문서들을 고치려고 한다.
- **-g**: 이미 존재하는 문서고에 파일을 추가한다.
- **-m**: 일단 문서고에 파일들이 있으면 그것들을 본래위치에서 지우고 문서고안으로 파일들을 옮긴다.
- **-r**: 문서고에 보조등록부에 있는 모든 파일들을 추가하면서 등록부들과 함께 련속적으로 작업한다.
- **-s**: 문서고에 체계 및 숨은파일들을 넣는다.
- **-u**: 파일이 현재 문서고안에 있는 파일보다 더 새것이지만 문서고안에 없으면 이미 존재하는 문서고에서 항목들을 교체한다.

주의 　**zip** 지령은 프로그람이 동작할 때 드물게 쓰이고 효률적인 많은 다른 선택항목들을 제공한다. 더 상세한것에 대해서는 **zip** man 페지를 참고하여야 한다.

zipgrep

설명 　**egrep** 를 써서 ZIP 문서고에 있는 하나이상의 파일에 대한 패턴을 탐색한다. 파일을 지정하지 않으면 문서고의 모든 파일을 탐색한다.

문법 　**zipgrep [egrep 선택항목] 패턴 zip 파일 파일…**

주요기발과 선택항목 　없음

주의 　**egrep's** 와 **zipgrep's** 의 패턴문법의 완전한 설명에 대하여서는 **egrep** 지령을 참고해야 한다.

zipinfo

설명

암호화상태, 압축형태, 문서고를 만드는데 리용된 조작체계 등을 포함하는 zip 문서고에 대한 상세한것을 현시한다. 기정으로 문서고의 매 파일에 대한 정보는 분리선우에 렬거된다. Zip 파일이 지정되지 않으면 표준입력이 처리된다.

문법

zipinfo [-12lM] zip **파일**[.zip] [**파일** …]

주요기발과 선택항목

- -1: 매 분리선우에 파일이름들만 렬거한다.
- -2: 머리부, 련결부, 주해부를 포함하는 파일이름들만 렬거한다.
- -l: 긴 형식(ls -1과 류사하게)으로 정보를 렬거한다.
- -M: more와 비슷한 모양으로 한번에 한페지씩 정보를 현시한다.

zmore

설명

more 지령에서와 거의 같은 방법으로 탐색하면서 한번에 한화면씩 압축된 본문파일의 내용을 현시한다. 파일이 지정되지 않으면 표준입력이 사용된다.

문법

zmore [**파일** …]

주요기발과 선택항목

없음

znew

설명

compress(.Z 파일)로 압축된 파일을 gzip(.gz 파일)가 사용한 형태로 변환한다. 파일이 지정되지 않으면 표준입력을 처리한다.

문법

znew [-ft9K] [**파일**.Z …]

주요기발과 선택항목

- -9: 가장 좋지만 가장 느린 압축방법이다.
- -f: .gz 파일이 이미 존재할 때조차도 압축한다.
- -K: .Z 파일이 새로운 .gz 파일보다 더 작다면 .Z 파일을 보관한다.
- -t: 원래의 .Z 파일을 지우기전에 새로운 .gz 파일을 검사한다.

부록 4. GNU 일반공개사용허가

1991년 6월 판본2

소 개

거의 모든 쏘프트웨어에 대한 사용허가들은 마음대로 공유하고 변경할수 없게 설계되였다. 그러나 GNU 일반공개사용허가는 모든 사용자들이 무료쏘프트웨어를 마음대로 공유하고 변경할수 있는 담보를 준다. 이 일반공개사용허가는 대부분의 무료쏘프트웨어협회 쏘프트웨어와 저자들이 승인한 프로그람에 적용된다(일부 무료쏘프트웨어협회 쏘프트웨어는 GNU 서고일반공개사용허가대신에 리용된다.). 당신의 프로그람들에도 이것을 적용할수 있다.

우리가 말하는 무료쏘프트웨어는 가격측면에서가 아니라 자유로운 사용측면이다. 일반공개사용허가는 무료쏘프트웨어의 복사본을 무료로 배포하고(당신이 바란다면 이 봉사를 맡아할수 있다.) 필요한 원천코드를 입수하거에서 얻을수 있으며 그 쏘프트웨어를 변경하거나 무료프로그람에 있는 부분을 리용하고 이러한 작업을 할수 있다는것을 알도록 설계되였다.

소유권을 보호하기 위하여 우리는 그 누군가에 의해서 당신이 이 소유권들을 거절당하거나 포기하지 못하도록 제한을 준다. 이러한 제한으로 해서 쏘프트웨어의 복사본들을 배포하거나 그것을 수정하면 책임을 져야 한다.

실례로 이러한 프로그람의 복사본을 무료나 유상으로 배포한다면 당신이 가지고 있는 모든 소유권을 접수자에게 주어야 한다. 당신은 그들이 원천코드를 입수하거나 얻을수 있게 하여야 한다. 그리고 그들이 자기의 소유권을 알수 있도록 이러한 조항들을 그들에게 보여 주어야 한다.

우리는 두 측면에서 당신의 소유권을 보호한다. (1) 쏘프트웨어에 대한 판권을 가지며 (2) 그 쏘프트웨어를 복사하고 배포하고 수정할수 있는 법적승인을 가지는 사용허가를 제공한다.

또한 우리는 매 저자와 우리자신을 보호하기 위해서 모든 사람들이 이 무료쏘프트웨어에 대한 담보가 없다는것을 리해하고 있는가를 확인해 보려고 한다. 쏘프트웨어가 누군가에 의해서 수정되거나 통과되는 경우에 우리는 다른 사람들에 의해서 공개된 문제들이 초기저자의 명성에 영향을 주지 않도록 입수한 사람들이 가지고 있는것이 원래의것이 아니라는것을 알기 바란다.

마지막으로 무료프로그람은 쏘프트웨어특허에 의하여 항상 위협을 받게 된다. 우리는 실제로 프로그람소유권을 만드는 무료프로그람재배포자들이 개별적으로 특허허가를 얻을수도 있다는데 대하여 주목을 돌려야 한다. 이것을 보호하기 위하여 우리는 임의의

특허가 모든 사람들의 무료사용을 위해 사용허가되거나 전혀 사용허가되지 말아야 한다는것을 명백히 하였다.

복사, 배포, 수정을 위한 조항과 조건은 다음과 같다.

복사, 배포, 수정을 위한 조항과 조건

0. 이 사용허가는 일반공개사용허가협약밑에서 배포될수 있다고 말한 저작권보유자의 통보를 포함하는 프로그람이나 다른 작업에 적용된다. 아래에서 《프로그람》이란 말은 임의의 프로그람이나 작업을 의미하며 《그 프로그람에 기초한 작업》이란 저작권법하에서 프로그람이나 그로부터 파생된 작업의 어느 하나를 의미한다. 즉 그것은 프로그람이나 그의 부분을 포함하는 작업을 말하는것이다. 직역이나 수정, 다른 언어로 번역 등이 이 작업에 속한다(이하에서 번역은 《수정》이라는 말로 제한없이 포함된다.). 매개 사용허가는 《당신》이라는 말로 부른다.

복사, 배포, 수정을 제외한 다른 활동들에 대해서는 여기에서 취급하지 않는다. 그것들은 이 령역밖에 있다. 프로그람을 실행하는 동작은 제한되지 않는다. 그리고 프로그람으로부터의 출구는 프로그람에 기초한 작업을 그의 내용들이 선택할 때에만 진행한다.

1. 당신은 매 복사에서 적합한 저작권통보와 담보의 거부가 현저하게 그리고 적당하게 공개된 프로그람원천코드의 원상그대로의 복사본들을 복사하고 배포할수 있다.

당신은 복사물을 전송하는 물리적동작을 위하여 료금을 물어야 하고 당신의 항목에서 료금대신 담보보호를 제공할수 있다.

2. 이 모든 조건들을 만족시키는 조건에서 우에서 지적한 1 절의 항목들에 맞게 프로그람의 사본이나 사본들 또는 그것들의 임의의 부분을 수정하여 그 프로그람에 기초한 제품을 만들수 있고 그것을 수정하거나 복사하거나 배포할수도 있다. 당신은 수정된 파일들이 당신에게 있는 파일들과 변경시간이 다르다는것을 가리키는 유명한 통지들을 나르게 해야 한다.

① 파일을 변경시켰다는것과 변경날자를 명백히 알수 있도록 변경된 파일들에 알림문을 포함시켜야 한다.

② 전체적으로 또는 부분적으로 프로그람이나 그의 임의의 부분을 포함하거나 그로부터 만들어 진 배포 또는 인쇄하는 임의의 제품들이 이러한 허가항목에 따라 모든 세번째 단체들에 무료로 총체적인 허가를 받도록 하여야 한다.

③ 일반적인 방법으로 그러한 대화형리용을 위해 실행을 시작하게 될 때 그것이 적당한 저작권통보와 담보가 없고(그렇지 않으면 담보를 제공한다는것을 알려 주는)사용자들이 이러한 조건들하에서 프로그람을 재배포할수 있다는 통보들을 포함하며 사용자에게 이 허가의 사본을 어떻게 보겠는가른 알려 주는 통지문을 인쇄 또는 현시하게 하여야 한다(례외: 만일 프로그람자체가 대화형이지만 보통 그러한 통지문을 인쇄하지 않으면 그 프로그람에 기초한 제품이 통지문을 인쇄하는것이 요구되지 않는다.).

이러한 요구들은 총체적으로 수정된 제품에 적용된다. 만일 그 제품의 동일하다고 볼수 있는 부분들이 그 프로그람으로부터 유도되지 않고 그자체를 독립적이고 개별적인 제품들로 볼수 있는 부분들이 타당성이 있으면 이 허가와 그 항목들은 그러한 부분들을

개별적인 제품들로 배포할 때 적용되지 않는다. 그러나 같은 부분들을 그 프로그람들에 기초한 제품인 전체의 부분으로 배포할 때 전체의 배포는 이 허가의 항목들에 기초하여 진행되여야 하며 다른 허가소유자들에 대한 이것의 허락들은 전체에 확장되여 그것을 누가 작성했든 관계없이 매 부분에 적용된다.

따라서 어떤 사람에 의해 총체적으로 작성된 제품에 대해 권리를 요구하거나 권리를 론박하는것이 이 절의 목적이 아니라 그 프로그람에 기초하여 유도 또는 수집된 제품들의 배포를 조종하는 권리를 행사하는것이다.

또한 이 허가에서 그 프로그람에 기초하지 않는 다른 제품을 그 프로그람(또는 그 프로그람에 기초한 어떤 제품)과 기억 또는 배포매개물로써 합쳐서 다른 제품을 만들지 못한다.

3. 다음의 어느 하나를 아는 조건에서 우의 1과 2의 항목에 따라 프로그람을 목적코드나 실행형식으로 복사하거나 배포할수 있다.

① 그것에 기계가 읽을수 있는 완전히 대응한 원천코드를 덧붙이는데 이것은 쏘프트웨어교체를 위해 리용되는 매개물로 우의 1 과 2 의 항목에 따라 배포되여야 한다.
② 그것에 적어도 3 년동안 유효한 서면으로 된 제공을 덧붙여야 하는데 이것은 임의의 해당한 원천코드의 기계가 읽을수 있는 완전한 사본을 제공하며 쏘프트웨어교체를 위하여 보통 리용되는 매개물로 우의 1과 2의 항목에 따라 배포된다.
③ 그것에 제공에 관하여 받은 정보를 덧붙여 해당한 원천코드를 배포한다(이러한 방법은 비상업적인 배포를 위해서만 그리고 우의 ②에 따라서 그러한 제공과 함께 프로그람을 목적코드 또는 실행형식으로 받았을 때에만 허용된다).

제품에 대한 원천코드는 그에 대한 수정을 쉽게 해주는 그 제품의 형식이다.실행형제품에 대하여 완전한 원천코드는 그것이 포함하는 모든 모듈들에 대한 모든 원천코드, 임의의 련관된 대면부정의파일들 그리고 실행형의 번역과 설치를 조종하는데 리용되는 스크립트들을 의미한다. 그러나 특수한 례외로써 배포된 원천코드는 기본성분들(번역기, 핵심부 등등)자체가 실행형을 동반하지 않으면 보통 실행형이 실행되는 조작체계의 그 성분들과 함께 배포되는(원천형식으로 또는 2 진형식으로)임의의것을 포함할 필요가 없다.

만일 실행코드 또는 목적코드의 배포가 지정된 곳으로부터 사본에 호출을 제공함으로써 진행된다면 3 자가 원천을 목적코드와 함께 복사하게 강요되지 않지만 같은 곳으로부터 원천코드를 복사하기 위하여 등가의 호출을 제공하는것은 원천코드를 배포하는것으로 된다.

4. 이 허가에 명백히 부합되는 다른 방법으로 프로그람을 복사, 수정,허가넘기기 또는 배포할수 없다. 다른 방법으로 프로그람들을 복사, 수정,허가넘기기 또는 배포하려는 임의의 시도는 이 허가에 따라 무효로 되며 권리들을 받은 단체들은 요구들을 충분히 지키는 한 자기들의 허가를 계속 가진다.

5. 이 허가에 수표하지 않았으면 그것을 받아 들일 필요는 없다. 그러나 이렇게 되면 프로그람이나 그로부터 유도된 제품을 수정하거나 배포할수 있는 허가를 받지 못한다. 만일 이 허가를 받아 들이지 않으면 이렇게 하는것은 법에 의해 금지된다. 따라서 프로

그람(또는 그에 기초한 임의의 제품)을 수정하거나 배포하여 이 허가를 받아 들이면 그렇게 한다는것과 프로그람이나 그에 기초한 제품들을 복사, 배포 또는 수정하기 위한 모든 항목들과 조건들을 나타낸다.

6. 프로그람(그에 기초한 임의의 제품)을 재배포할 때마다 수신자는 자동적으로 초기허가자로부터 이 항목들과 조건에 따라 프로그람대상들을 복사,분배 또는 수정할수 있는 허가를 받는다. 여기서 인정된 수신자의 권리리행에 대해 그 어떤 제한을 더 부여할수 없다. 이 허가에 따라 3자들이 동의하도록 강요할수 없다.

7. 만일 법정판별의 결과 혹은 전매품위반의 진술, 또다른 리유(전매품결과에 제한되지 않는다.)로써 조건들이 이 특허의 조건들을 위반한다고 당신을 (법정지령 또는 협정, 기타들에 의하여) 강요한다면 그것들은 이 특허의 조건들로부터 당신을 변호하지 못한다. 만일 사람들이 배포하여 이 허가에 따르는 의무와 임의의 다른 적합한 의무를 동시에 만족시킬수 없다면 프로그람을 배포할수 없다. 실례로 특허권허가가 사용자를 통해 직접적으로 또는 간접적인 사본들을 받는 모든 사람들이 특허권사용료를 물지 않고 프로그람을 재배포하는것을 허가하지 않으면 사용자가 그것과 이 허가를 둘다 만족시킬수 있는 유리한 방법은 프로그람의 배포를 완전히 그만두는것이다.

이 항목의 임의의 부분이 어떤 특수한 환경하에서 무효로 되거나 강요되지 않으면 항목의 균형 맞추어 지고 다른 환경들에서 항목이 전체적으로 적용되게 된다. 사람들이 임의의 특허권들이나 재산권리요구들을 침해하게 하거나 또는 그러한 임의의 요구들의 정당성을 론박하는것이 이 항목의 목적은 아니다. 이 항목의 목적은 공공허가실천에 의해 리행되는 자요쏘프트웨어재배포체계의 완전성을 보호하는것이다.

많은 사람들은 그 체계의 시종일관한 응용을 믿고 그것을 통하여 배포되는 광범한 쏘프트웨어에 많은 기여를 하고 있다. 임의의 다른 체계를 통해 쏘프트웨어를 배포하려고 하는가를 결심하는것을 저자나 기증자의 의무이고 허가는 그러한 선택을 강요할수 없다. 이 항목은 이 허가의 나머지부분의 결과가 어떻게 되는가를 확인할수 있게 한다.

8. 프로그람의 배포나 리용이 특허권이나 저작권을 받은 증계물에 의해 어떤 나라들에 국한되면 프로그람을 이 허가의 영향하에 넣은 원래의 저작권소유자는 그러한 나라들을 제외한 명백한 지역적배포한계를 추가하여 제외되지 않은 나라들에서만 배포를 할수 있게 한다. 그러한 경우에 이 허가는 마치도 한계가 그안에서 작성된것처럼 적용한다.

9. 무료쏘프트웨어재단은 언제나 일반공개허가의 수정된 또는 새로운 판본들을 출판할수 있다. 그러한 새로운 판본들은 현재의 판본들과 류사하지만 구체적으로 새로운 문제들이다.

매 판본에는 구별되는 판번호가 주어 진다. 프로그람이 자기와 《임의의 이후판》에 적용되는 이 허가의 판번호를 지정하면 자유쏘프트웨어재단에 의해 출판된 그 판본이나 임의의 이후판본의 항목과 조건들에 따라서 선택한다. 만일 프로그람이 이 허가의 판번호를 지정하지 않으면 자유쏘프트웨어재단에 의해 출판된 임의의 판본을 선택할수 있다.

10. 프로그람의 부분들을 그의 배포조건들이 다른 무료프로그람에 포함시키려고 하면 저자에게 서면으로 허가승인을 받아야 한다. 자유쏘프트웨어재단에 의해 저작권이 담보되는 쏘프트웨어에 대해서는 무료쏘프트웨어재단에 승인을 받아야 한다. 때때로 이것에 례외가 있다. 이렇게 하게 되는것은 두가지 목적때문인데 하나는 무료쏘프트웨어로부터 유도된 모든것들이 무료로 봉사되게 하자는것이로 다른 하나는 일반적으로 쏘프트웨

어의 공유와 재리용을 장려하자는것이다.

담보 없음

11. 프로그람이 무료로 봉사되도록 허가되기때문에 응용법에 의하여 허락되는 한계에서는 프로그람에 대하여 담보가 없다. 그렇지 않다고 서면으로 표시된 경우를 제외하고 저작권소유자들이나 다방들은 표시되거나 내포된 임의의 종류의 담보도 없이 프로그람 “ASIS”를 제공하는데 이 담보는 상업적리용과 특별한 목적에 관한 내포된 담보들을 포함하지만 그에 국한되지는 않는다. 프로그람의 질과 성능에 관하여 제기되는 모든 문제들은 사용자가 책임진다. 프로그람에 결함이 있다고 하면 사용자가 모든 필요한 봉사, 수리와 정정을 위한 비용을 부담한다.

12. 응용법에 의해 요구되지 않거나 서면으로 합의되지 않는 한 우에서 허락된대로 프로그람을 수정하거나 배포할수 있는 임의의 저작권소유자나 타방은 있을수 있는 손상들에 대하여 알고 있었더라도 그에 대하여 사용자에게 그 어떤 경우에도 책임을 지지 않는다. 그러한 손상들은 프로그람(자료손실, 정확치 않게 전달되는 자료, 사용자나 3자에 의한 손실 또는 임의의 다른 프로그람과 함께 동작하지 않는 고장을 포함하지만 그에 국한되지 않는)을 리용할 때 생기거나 리용할줄 몰라서 생기는 임의의 일반적인 ,특수한, 우연적인 또는 결과적인 손상들을 포함한다.

조항들과 조건들의 끝

이러한 조항을 프로그람에 어떻게 리용하는가

만일 당신이 새로운 프로그람을 개발한다면 그리고 그것이 대중에게 사용되는 가장 큰 가능성을 가질것을 바란다면 이것을 성취하기 위한 가장 좋은 방법은 그것을 누구나 다 이러한 조항들하에서 재배포와 변화시킬수 있는 무료쏘프트웨어를 만드는것이다.

그렇게 하려면 프로그람에 다음과 같은 통지를 붙여야 한다. 담보의 배제를 가장 효과적으로 전달하기 위하여 매 원천파일의 시작에 그것들을 붙이는것이 가장 안전하다. 매 파일은 적어도 “copyright”행과 완전한 통지를 찾을수 있게 하는 지시자를 가지고 있어야 한다.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
This program os distributed in the hope that it will be useful,
But WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY of FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
You should have received a copy of the GNU General Public License

Along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

또한 당신이 전자적으로 종이우편에 의하여 어떻게 련계를 맺겠는가에 대한 정보를 첨가한다.

만일 프로그람이 상호작용하는것이라면 그것은 그것이 호상작용하는 방식으로 시작할 때 다음과 같이 짧은 통지를 내보낸다.

Gnomovision version 69, Copyright © 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.

This is free software, and you are welcome to redistribute it
Under certain cpndotions; type 'show c' for details.

가정지령 즉 "show w"와 "show c"는 일반공개특허의 부분들을 정확하게 보여 주어야 한다. 물론 사용하는 지령들은 "show w"와 "show c"와 다른것으로 호출될수 있다. 그것들은 항상 프로그람에 알맞게 마우스찰칵이나 안내항목들까지도 될수 있다.

만일 당신이 자기의 직업(당신이 프로그람작성자로 일한다면)이나 학교를 주어야 하는데 프로그람을 위하여 필요하다면 ` "copyright disclainer"표식을 한다. 여기에 실례가 있는데 이름들을 바꿀수 있다.

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989

Ty Coon, President of Vice

이 일반공개사용허가는 당신의 프로그람이 소유된 프로그람들과 결합되는것을 허용하지 않는다. 만일 당신프로그람이 보조루틴서고라면 그것을 서고를 가지고 있는 소유된 프로그람들을 련결하도록 허용하는것으로 생각할수 있다. 만일 이것이 당신이 하고 싶은것이라면 이 특허대신에 GNU서고일반공개사용허가를 리용할수 있다.

부록 5. 비인텔계 하드웨어에서의 Linux

Linux는 Intel x86과 Pentium처리소자들뿐만아니라 다른 각이한 하드웨어가동환경들에서도 리용할수 있다. 이 부록은 일부 비인텔계의 주요한 Linux 판본들에 대하여 개괄한다. 리용할수 있는 Linux 포구들에 대한 종합목록은 http://www.linux.org/projects/ports.html에 있는 하드웨어의 포구에 대한 대상과제의 Linux직결목록을 참고하면 된다.

Compaq Alpha 처리소자를 위한 Linux

Compaq 의 Alpha CPU 는 전형적인 RISC CPU 로서 시장에서는 매우 빠른 처리소자들중의 하나로 알려 져 있다.

Alpha CPU 를 위한 Linux 는 현재 리용할수 있는 비인텔계 하드웨어를 위한 Linux 의 가장 견고하고 안정하며 높이 평가된 판본들에 속한다. X Windows 로부터 Web 열람기를 위한 망들에 이르기까지 모든것이 잘 동작되며 지어는 Intel x86 용으로 개발된 많은 Linux 프로그람들을 실행할수 있게 한다. Alpha CPU 를 위한 Linux 에 대한 더 자세한 정보는 http://www.alphalinux.org를 참고할수 있다.

Alpha 를 위한 Linux 는 값 눅은 Universal Desktop Boxes 로부터 값 비싼 64bit CPU 체계에 이르기까지 수많은 Compaq Alpha 에 기초한 체계들에서 동작한다.

Red Hat 는 Alpha 처리소자를 위한 Red Hat Linux 의 판본을 제공한다.

매몰형 Linux

ELKS(Embedded Linux Kernel Subset)대상과제는 8086 및 80286 급의 낡은 PC, 손바닥형콤퓨터, 매몰형조종체계와 같은 매몰형체계들을 위한 Linux 판본을 만드는것을 목적으로 하고 있다. 이 대상과제로부터 사용할수 있는 Linux 의 현재판본은 PC 우에서 실행되며 완전한 체계구성에 400~512K RAM 을 요구한다.

이 판본은 완성되자면 아직 멀었다. 지금은 시동할수 있는 능력, 가상조종탁의 사용, 플로피디스크의 호출, 작은 프로그람들의 실행이 가능할뿐이다. 그리고 현대도형사용자대면부들의 특성을 보다 작은 장치들에서 실현할수 있게 하는 극소형창문들을 개발하는 작업이 진행중에 있다.

이것은 체계개발이지만 다양한 하드웨어제한밑에서 Linux 가 실행될수 있는 유연성측면을 강조한다. ELKS 대상과제는 http://www.elks.ecs.soton.ac. uk/에서 볼수 있다.

Motorola 680x0 처리소자를 위한 Linux

Linux/m68k 는 Amiga, Atari, Macintosh 콤퓨터들에서 볼수 있는 Motorola 68020 ~ 68060 CPU 에서 실행될수 있게 설계된 Linux 의 포구이다.

Linux/m68k 의 견고하고 안정한 공개물들은 현재 다음의 체계들을 포함하는 Amiga, Atari, Macintosh, NeXT 콤퓨터들에서 사용할수 있다.

- Amiga A2500
- Amiga A3000
- Amiga A3000T
- Amiga A4000/040
- Amiga A4000T/040
- Amiga A4000T/060
- Amiga DraCo(계렬)
- Amiga BoXeR(계렬)

- Atari Falcon
- Atari Falcon with AfterBurner 040
- Atari TT
- Atari Medusa(계렬)
- Atari Hades(계렬)
- HP9000/300 워크스테이션
- Macintosh II(각이한 모형)
- Macintosh SE/30
- Macintosh LC(각이한 모형)
- Macintosh Performa(각이한 모형)
- Macintosh Quadra(각이한 모형)
- Macintosh Centris(각이한 모형)
- NeXT 워크스테이션
- Q40(Sinclair 파생)
- Sun 3 워크스테이션(3/50, 3/60, 3/160)

MIPS 처리소자를 위한 Linux

Linux/MIPS 는 다음의것을 포함하는 MIPS 처리소자들을 위한 대부분의 체계들에서 실행될수 있게 설계된 Linux 의 포구이다.

- Acer PICA-61
- Compaq DECStation 5000/2x, 5000/100, 3100
- MIPS Magnum 4000PC
- Olivetti M700-10

현재공개물은 안정성이 고려되였으며 일련의 콤퓨터들에 주변장치들을 적재하는데서 일부 불충분한 점들이 있으나 망과 NFS 를 지원한다. 아주 최신은 아니지만(현재판본은 Linux 핵심부 2.1.73 에 기초하고 있다.) 이 대상과제의 Web 싸이트에 있는 쏘프트웨어로부터 완전한 배포물을 얻을수 있다. Linux/MIPS 의 Web 싸이트는 http://lena.fnet.fr/이다.

Power Macintosh 를 위한 Linux

MkLinux 는 Power Macintosh 계렬 콤퓨터들에서 실행되고 있는 Open Group Mach 의 극소핵심부우에서 Linux 를 실행한다. 극소핵심부의 구성방식은 극소핵심부 그 자체의 이식에 집중하면서 조작체계의 이식을 쉽게 할수 있게 설계되였다.

Apple 콤퓨터는 MkLinux 대상과제를 완전히 지원한다. 이 대상과제의 싸이트는 http://www.mklinux.org 이다. 완전한 배포물은 현재 사용할수 있는 직결문서 혹은 CD-ROM 우에서 리용할수 있다.

Power Macintosh 에 Linux 포구를 형성하는 다른 대상과제들로서는 PowerPC Linux 배포물(이 부록의 마지막에 있는 PowerPC 처리소자에 대한 부분에서 론의되는)과 Powermac/Linux(http://www.cs.wisc.edu/~tesch/ linux_info/)가 있다. 이 두개의 대상과제들은 극소핵심부우에 놓이지 않는 PowerPC 처리소자(Power Mac 체계들의 심장부로 되는)에 대한 Linux 의 판본들이다. 이러한 설계방식은 다른 하드웨어에 이식하기는 어렵지만 성능을 높일수 있는 좋은 점을 가지고 있다. 독특한 대상과제의 Powermac/Linux 코드는 PowerPC 를 위한 Linux 의 판본을 만들기 위하여 PowerPC Linux 가 개발한 LinuxPPC 원천코드와 결합된다.

Apple 은 그의 최신판 Mac OS X 조작체계에 Darwin 이라고 불리우는 Linux 의 판본을 확장하였다. Mac OS X 핵심부는 두개의 서로 다른 Unix 계렬(Carnegie-Mellon 의 Mach 3.0 과 Berkeley Standard Distribution 판본 4.4)들에 기초하고 있지만 여전히 실제상 많은 Linux 쏘프트웨어를 포함하고 있다.

PowerPC 처리소자를 위한 Linux

PowerPC 처리소자계렬우의 Linux 를 지원하는데서 빠른 발전이 이룩되였다. LinuxPPC 대상과제(http://www.linuxppc.org/)는 PowerPC Linux 에 의해서 관리되고 개발된다. 그것은 Power Macintosh 콤퓨터, Amiga 의 Power-UP 체계, Be 회사의 BeBox, PowerPC 에 기초한 RS/6000, PreP, IBM 의 CHRP, Umax 의 워크스테이션 등의 넓은 범위에서 실행되는 안정하고 완전한 배포물을 제공한다.

지원되는 응용프로그람들의 특성들은 다음과 같다.

- PCI 모선체계
- SCSI 와 IDE 하드디스크구동기
- 다매체지원
- Netscape 와 같은 Web 열람기
- X Windows 와 넓은 범위의 창문관리기
- Java JDK 1.2.2

대상과제는 Wine 이 Intel x86 CPU 우에서 Linux 를 위한 Windows 모방프로그람을 개발하려고 시도하는것처럼 PowerPC 의 Linux 내에서 실행되는 Mac OS 의 모방프로그람을 개발하는 중에 있다. 그의 일부 결과가 최근에 공개된 Mac OS X 이다.

SPARC 처리소자를 위한 Linux

한때 S/Linux 로 알려 졌던 UltraLinux(http://www.ultralinux.org/)는 Sun 의 SPARC 와 UltraSPARC 처리소자들에 Linux 를 이식하기 위한 대상과제이다. 현재 그 포구는 아주 안정하며 가장 새로운 64bit UltraSPARC 에 기초한 워크스테이션들도 지원한다. 이 책에서 보는바와 같이 Slackware 와 Debian 은 가장 최근배포물에 대한 PARC 판본을 가진다.

색　인

ㄴ

ㄷ

ㅈ

ㅊ

ㅋ

ㄷ

ㅍ

*　*　*

A

D

E

F

G

H

I

M

N

O

S

T

U

V

W

X